M 2920 Pete Eaton (WB9FLW,2970) 10/12/85 3:08 PM L:125
KEYS:/COMPUTER TRADER MAGAZINE (CTM) ANNOUNCES PACKET SWEEPSTAKES/
TO: (Group 95), WB9FLW

To: All
From: Pete WB9FLW
Subject: COMPUTER TRADER MAGAZINE (CTM) PACKET $weepstakes Rules
Dist: Open
Note: Below is the exact text of a flyer distributed by CTM

In an effort to promote Packet Radio, CTM is sponsoring a ham contest like nothing you've ever seen. This contest is open to all licensed amateurs worldwide and boasts a prize list such has never been offered before anywhere!

In actuality, there are two separate contests involved in the CTM Packet $weepstakes. The first is what we affectionately refer to as the CTM Packet 50, which consists of connecting to all 50 U.S. States. The second is called the CTM Packet 100, which involves connecting to 100 or more countries.

The rules are simple. We don't want to play keep away with these prizes, we feel that it's going to be hard enough as it is.

1. All contacts must be made using packet radio on any legal amateur frequency

2. No store and forward contacts will be allowed. This means that you cannot use BBS or the store and forward capabilities of PACSAT to make a valid connection. All contacts must be live two way communications. Any number of digipeaters or gateways will be permitted.

3. All contacts must be made from the same state and country. We wouldn't want somebody traveling to all these places just to get the goodies now, would we?

4. All contacts must be confirmed via standard QSL type cards. All QSL cards must be signed by their operator, and contain the following information:

   Date and time of QSO (preferably in GMT).
   Operator's name, address, city, state, zip code and county.
   Must state that the contact was made using packet radio.
   Manufacturer of Terminal Node Controller used to make the connection (his TNC, not yours).

5. All contacts must take place after 0000 GMT on December 1, 1985.

6. Winner will be the earliest postmarked entry received that meets the contest criteria. Keep in mind that the Packet 50 and the Packet 100 are two seperate contests.

7. An entry consists of all the QSL cards needed to verify your entry (50 or 100 depending on which contest you are working on), along with your own signed QSL card and your telephone number. Send your entry to:

   CTM Packet $weepstakes
   Contest Editor
   1704 Sam Drive
   Birmingham, AL 35235

   Please do not send in your entry until you have completed all of the requirements for a particular contest. QSL cards will be returned only if you provide an SASE for them. You will be notified when your entry has been verified by our staff.

8. There is no ending deadline for this contest. The contest will continue even after all prizes have been awarded. A handsome certificate that you can be proud of will be awarded to all who submit valid entries. Each month, results of the contest, as we have them, will appear in the contest section of CTM.

9. Prizes will be awarded at the next Dayton Hamvention following verification of the entry. In the event that you are unable to attend the Dayton Hamvention, other arrangements will be made. At this time, the prize list is still growing. As it sits right now, this is way the prizes will be distributed. This prize list will be finalized in the January 1986 issue of CTM.

CTM PACKET 50

First Prize: Icom state of the art 2 meter rig (currently the IC-271H with AC supply)

Second Prize: Heathkit HD-4040 Terminal Node Controller

Third Prize: Kantronics Terminal Node Controller

Fourth Prize: GLB PK-1 Terminal Node Controller

Fifth Prize: AEA PK-64 Commodore 64 Terminal Node Controller and your choice of software program

CTM PACKET 100

First Prize: ICOM state of the art HF station (currently the IC-751 dream station with the IC-2KL amplifier and the IC-AT500 antenna tuner)

Second Prize: Kantronics Terminal Node Controller

Third Prize: Heathkit HD-4040 Terminal Node Controller

Finally, we would like to make a minor request from all contest participants. Please drop us a line telling us that you are participating in the contest and from time to time drop us another card or letter letting us know how well you're progressing as well as any comments or suggestions that you may have. Please send these cards and letters to the address mentioned above for entries. As we receive information from you, we will print it here in the contest section for all to see. Good Luck!

-Signed-
Chet Lambert, W4WDR
Editor/Publisher

-Signed-
Jim Griffith, WA5RAX
Packet Editor

-Signed-
Glenn Popiel, WA4FTX
Contest Editor

For additional copies of the contest rules, send a self addresses stamped envelope (SASE) to: CTM Packet $weepstakes, Contest Editor, 1704 Sam Drive, Birmingham, AL 35235.

M 2923 Pete Eaton (WB9FLW,2970) 10/12/85 3:18 PM L:18
KEYS:/I'VE GOT AN UNEASY FEELING ABOUT THIS/
TO: (Group 95), WB9FLW

To: All
From: Pete WB9FLW
Subject: I've got an uneasy feeling about this

The announcement by Computer Trader Magazine of a Packet Contest is one respect is exciting, that of promoting Packet Radio. However in another light I've got an uneasy feeling about it, stated simply:

> "Can you have an on-the-air contest, the winner to receive merchandise as a prize?"

A couple of us at the ARRL convention (were the flyers were handed out) had the same question, anybody got an answer?

Oh yes the distribution on this is -OPEN-, I'm sure it will generate some "packets-in-flight"

M 3628 John Gates -NAPRA- (N7BTI,2988) 10/15/85 2:13 AM L:5
KEYS:/SCATTER/6 METER/

Am looking (on behalf of WA7ETE, Steve here in Seattle) for anyone interested in working packet on 6m meteor scatter. Steve wants folks about 1000 miles from Seattle. If you are interested or know of anyone

please let me know on DRNET.
Regards, John

M 3128 Skip Hansen (WB6YMH,2964) 10/13/85 12:46 PM L:32
KEYS:/WESTNET GOES EAST/
TO: (Group 95), ARRL

Jeff,

Westnet has finally made it out of the state! Two Digipeaters, WA7HXO-1 and K7WS, have been recently installed on Mount Potosi located southwest of Las Vegas at 6000 feet. They link Los Angeles to Las Vegas thru the AA6TN-1 digipeater in the Big Bear Lakes area, all on 145.01. Extending the path thru existing digipeaters to Sacramento would look like:

K7WS -> AA6TN-1 -> NK6K-1 -> W6AMT-2 -> W6AMT-1 -> WA6RWN -> W6BXN -> K6QIF

The following is forwarded from K7WS:

THE K7WS DIGI WAS INSTALLED ON THE 28TH OF SEPT.
SITE ELEVATION APROX. 6000 FEET ASL
THE EQUIPMENT AT K7WS IS A MOTOROLA METRUM II RUNNING 15 WATTS
ANTENNA 6DB STATION MASTER AND TNC IS A GLB.
NO SPECIAL MODS , RUNNING GLB'S SOFTWARE
WE WOULD LIKE TO GO 220 AND 9600 BAUD HERE IN VEGAS AND HAVE
POTOSI FOR A LINK SITE
PLAN: 145.01 DIGIS ON POTOSI, ANGELS, GOLD MTN (NEAR GOLDFIELD), AND
CHRISTMAS TREE PASS (NEAR LAUGHLIN, NV)
THE WHOLE SOUTHERN 1/2 OF THE STATE OF NV AND PARTS OF UTAH, AZ AND CA
WILL HAVE COVERAGE WITH THESE DIGIS
PLANS TO LINK TO PHOENIX,SALT LAKE AND RENO ARE IN THE WORKS
THERE WERE 6 PACKETEERS IN VEGAS WHEN THE DIGI WENT ON THE AIR...
NAME OF GROUP: NEVADA AMATEUR RADIO ASSOCIATION OF LAS VEGAS, INC.
BY WAYNE SCHENK (K7WS) AND RANDY CURRAN (WA9PPD).

I hope this will be useful information for Gateway. 73's Skip WB6YMH.

M 3194 Phil R. Karn (ka9q,2979) 10/13/85 6:03 PM L:20
KEYS:/PACKET CONTESTING/
TO: (Group 95), ka9q

I see a fatal flaw in the contest rules themselves, ignoring the legality of awarding prizes for amateur operating.

The rules outlaw "store and forward" messages, but they permit the use of digipeaters. What is a digipeater if it isn't a store-and-forward message relay device? The only difference is that it doesn't hold as much at one time, and it only transmits it once. Other than that, they're conceptually the same.

I am tired of short-term "promotions" such as this one which do very little to promote the state of the art of amateur packet radio (as opposed to the "hype" and QRM levels). If somebody's going to sponsor a contest, why not put a little technical challenge into it? How about the first 1000-mile QSO conducted entirely with K9NG modems? How about the first QSO conducted over a network with automatic routing? And so forth.

If you're going to put up with contests, at least they ought to have a constructive purpose.

Phil

M 3237 Lyle Johnson (WA7GXD,2973) 10/13/85 10:25 PM L:7
KEYS:/ONE MORE THOUGHT ON PACKET CONTESTS/
TO: (Group 95), WA7GXD

One more thought for the %$^&$^& packet contest.

How about "no weekend contacts allowed" with a weekend being defined as noon thrusday through noon monday. The friday should be New Zealands and the Monday should be Hawaiis.

Lyle

M 3295 Jack Brindle (WA4FIB,2963) 10/14/85 2:05 AM L:41
KEYS:/SOAPBOX REQUESTS/
TO: (Group 95)

Greetings all Packeteers. I am Jack Brindle, WA4FIB, and have recently joined the ranks of those here on DRNET. I am still stumbling around here, but have managed to find a few things. In way of introduction, my background is communications engineering, both in Hardware and Software. I spent 3-1/2 years working with K4GFG at Motorola in Ft Lauderdale, and recently parted with Paradyne in Largo, Fl where I was in the Network design group. My specialty has become network communications. I am the author of the MacPacket series of programs for the Apple Macintosh.

My current work involves (among other things) the design of a session layer protocol for packeting, which will be unveiled at the Southnet Packetfest in November.

In reviewing the various DRNET conferences, I have found quite a bit of discussion on two of the three layer 3 routing methods, and discussion of TAPR's new NNC. However, I have not found a solid proposal for a Network layer design! In fact, the only "solid" proposal to be found in amateur packet literature is WB4JFI's modified X.25 layer 3 protocol published in the proceedings of the third ARRL Networking conference. Now, before Phil jumps on me for overlooking his TCP/IP proposal, let me note that it is in the form of "what we should do" instead of an actual protocol specification.

Now, I have my own preferences for just how the network should look, but, I believe that those who are arguing about the virtues of the various methods should stop, regroup, and come out with a solid specification detailing the layer 3 header information along with a complete set of implimentation notes so that we can proceed with writing code. There are programmers out now trying to code a layer 3, but with the only spec commonly available being Fox's incomplete spec (among other things missing are details on the implimentation of the very important priority handling proceedures), it is the target of most of these efforts.

There seems to be a lull in DRNET activities, I hope that can be interpreted as network designers writing the specs. If not, think of this one: We MAY be stuck with an unworkable network layer unless we get off our "pads" and get this info out!

Now I will climb down off my soapbox to hear responses of others. As I said before, I also have some ideas on how a packet network should be designed. Come to the Southnet Packet Conference in November to find out what and how!

73's to all, and I hope we start seeing some protocol specs appearing on the system. (If some are already here, please excuse my tirade and point me in the proper direction).
- Jack B. WA4FIB.

M 3299 Phil R. Karn (ka9q,2979) 10/14/85 2:28 AM L:37
KEYS:/"LEVEL 3"/
TO: WA4FIB, (Group 95)

Jack, the reason I did not put a complete protocol specification into my paper for the 4th proceedings is very simple: they have already been published in other forums, and since they are entirely suitable as-is for our needs I felt it necessary to include them only by reference. I have shipped out quite a few paper and electronic copies of the ARPA documents specifying IP, TCP and ICMP, and I have also recently received a paper copy of MIL-STD-1778, the "DoD rewrite" of the TCP spec. I will be shipping a copy of the latter to the ARRL Digital Committee (as I have with all my other documents) and I can sustain a VERY LIMITED capability to satisfy direct requests (SASEs are greatly appreciated, as these specs are a few hundred pages each..

The only specification which was necessary is the one describing the interface between IP (as layer 3) and AX.25 (as layer 2); this was discussed in detail in the paper and summarized in the appendix.
As of yet, however, I have received no comments from the Committee or anyone active in amateur packet radio, although I have received a few helpful suggestions from amateurs within the ARPA community who seem to have a latent interest in packet radio.

This seems to be a good time to mention that I am nearing completion of a "C"-language implementation of the TCP protocol for general purpose amateur packet radio use. Since one of the concerns about this protocol has been its complexity and implementation size (and whether it would fit on a TNC), I thought it would be interesting to give the current object-code size of all the modules making up TCP: 5K. That's right, FIVE kilobytes. That's all. If we are to do networking RIGHT (i.e., by avoiding "transport level bridges", this protocol belongs in each and every TNC. The "network controller" boxes can then concentrate on the jobs of packet switching and routing, which are sufficient challenges in themselves without having to support some kind of "user interface".

Further discussion on this subject should be moved to the "amateur radio level 3" conference.

73, Phil Karn, KA9Q

C2987 CC199 Skip Hansen (WB6YMH,2964) 10/13/85 10:00 PM L:27
KEYS:/MULTICONNECT TRANSPARENCY/

Howie,

I just finished wirewrapping a 2764/27128 programmer and it works! Now I can stay current without relying on the mail. I tested the programmer by burning the 1.1.1f code, just installed it and it fired right up.

I am very interested in going to multi-user on my host with two R.F. ports and multi-connect on each port. So far I haven't started writting any code, but have been giving the problem a lot of thought. The only problem area I see with the current configuration is transparency. I currently run my TNC in transparent mode to allow binary file transfers (I have had a binary file transfer program on line for over a year, but only two stations have used it in binary mode so far). I believe binary file transfer capability will become more important as time goes on. I would like to suggest that the TNC software be modified to handle stream switch characters received in incomming packets a bit differently than the present implementation. I would like to see the stream switch character simply doubled when it is sent over the serial port. This would preclude using "A","B","C",or "D" as the stream switch, but I think that would be an acceptable limitation! This would allow transparency with very simple software on the host end. It same technique could be used for sending characters to the TNC as well. If user=1 then all of this would be disabled of course. What do you think ?

73's Skip WB6YMH

C2987 CC200 Howard Goldstein (N2WX,2987) 10/14/85 7:47 PM L:38
KEYS:/16K RAM TNC 2/1.1.1/CKSUM $2F/+READ THIS ITEM/

To: Beta
Fm: Howie
Re: 16K RAM version 1.1.1g, ten connection, release description
Dist: Closed, for now

The .HEX of the release described hr can be had with +READ C2987 CC200.

TNC 2 software for 16K RAM is done. Buffer enlargement is one of the things addressed. Here are some other changes:

o - Up to TEN connections/links are allowed

o - A clock adjustment is installed. The associated command is CLKadj nnn, where 0=<nn=<65535. The adjustment effects the time by subtracting 9.1667ms every 100nn msec. Ex:
only WHEN CLKadj != 0,

$$\text{speed change of clock, in \%} = 100 - 9.16667 * \frac{1}{\text{CLKadj}}$$

o - *** DISCONNECT messages are stamped according to CONstamp

o - USER command is changed to USERS, and

```
     o - The USERS parameter has new significance:
          USERS = 0  allows incoming connections on any free stream
          USERS = 1  allows incoming connections on stream A only
          USERS = 2   "  "    "  "    "     "   on streams A & B
          USERS = 3   "  "    "  "    "     "   on streams A,B, & C
                    etc...
```

o - A bug where the output stream didn't change on ^C from CONV mode on a different stream has been fixed.

73 Howie

C2987 CC202 Howard Goldstein (N2WX,2987) 10/14/85   7:56 PM L:8
KEYS:/MULTI CONNECT TRANSPARANCY/
A: 199

Skip - That's one of the ways to do it, but it still wouldn't be failsafe when the remote sends your STReamsw a number of times in a row. (Escaping the escape which is escaping the next escape which is....)

I agree that converting them all to "$" does leave something to be desired as far as transparancy/finesse.

How 'bout the TNC sends a length byte before all that is data when some (as yet undetermined) command is set? I think Lyle suggested something like this a while ago... 73 Howie

C2987 CC203 Skip Hansen (WB6YMH,2964) 10/14/85 11:09 PM L:21
KEYS:/MULTICONNECT STRANGITIES/

Howie,

I was testing 1.1.1f with Harold last night and found an interesting characterstic. I was connected to myself thru a digipeater on stream A with user set to 4, when Harold attempted to connect to me he got a busy message and I saw a connect request message. When I switched to stream B and connected to him all worked ok. Streams C & D were in the disconnected state. AX25LVL2 was on and connect mode was transparent.

Another thing I ran into (which was basically an cockpit problem) was when I finished playing with multi-connect and put the host back on line I was active in stream B, I set user to 1 and forgot to reselect stream A. During this time the TNC would set garbarage beacon packets and ignore connection requests. When I reselected stream A all returned to normal. I might be worth while to force the selected stream into the legal range when user is changed, or prevent user from being changed until a legal stream is selected manually.

73's Skip WB6YMH

M 3970 Tom Clark (W3IWI,2976) 10/15/85 11:51 PM L:26
KEYS:/DUAL-PORT DIGIPEATER ON TNC2?/A MODEST PROPOSAL/
TO: (Group 95)

The following was sent to KE3Z @ W1AW via EASTNET and I thought I'd share it with you. John sent back a reply indicating he was interested in the idea and would take a look at it:

- - - - - - - -

John:

Came up with an idea I though I'd bounce off you re Dual-port digi (DPD). The thought began by noting that (a) X820's are starting to dry up, (b) the multiple power supply is a hassle, (c) you still have to add the state machine and modem for 1200 baud use and (d) the $ and time cost for integrating the 820, power supply, 1200 baud modem, state machine, 9600 baud modem, etc. into a case will deter many from getting DPD's on the air to help with our present congestion problems.

The critical parts of the TNC2 design look very much like an X820 and it has one state machine plus modem included as well as having been designed for lower power consumption. How much trouble would it be to port the X820 DPD code over to the TNC2? The present serial async port would have to have some minor mods (like bringing the clock lines to the DB25) to use the K9NG/KE3Z add-on boards for 9600 but the TNC2 looks like a better DPD "engine" than the 820 to me.

Thoughts?

73, Tom

M 4008 J. Gordon Beattie Jr. (N2DSY,2990) 10/16/85 1:28 AM L:5
KEYS:/KE3Z CODE ON ALL KINDS OF MACHINES !/
A: 3970 TO: (Group 95)

Tom et al:
I have ported the KE3Z code to the Protocom board I mentioned in earlier messages. It works fine. I can't see any major problem with porting Jon's code to the TNC2. 73, Gordon

M 4022 Lyle Johnson (WA7GXD,2973) 10/16/85 2:17 AM L:11
KEYS:/TNC 2 CABINET END PANELS ARE ENROUTE/
TO: WA7GXD, (Group 95)

Just in case you were wondering, the cabinet end plates came (finally!) and this time they actually fit !!!!

If you are among the lucky 511 folks who were sent a TNC 2 kit earlier you will be happy to note that the cabinet end plates were sent out to you by first class mail today (overseas by airmail).

With luck, the next lot of 100 to 300 kits should go out sometime next week and we should be caught up by the end of the following week.

Hang in there! TNC 2s are coming!

C2987 CC204 Howard Goldstein (N2WX,2987) 10/15/85 11:00 AM L:3
KEYS:/MULTICONNECT STRANGENESS/RELEASES 1.1.1A-F/
A: 203

The strangeness in USERS > 1 is corrected in the 16K RAM 1.1.1 (cksum $2F)

Releases 1.1.1a-f act like USER 1 when USER > 1. 73 Howie

C2987 CC205 Tom Clark (W3IWI,2976) 10/15/85 11:31 PM L:17
KEYS:/1.1.1G RUNNING AT IWI/MORE ON DATA "BLACK HOLE"/

Howie, got 1.1.1g (16k version) up and running in the Rev.2 board here with no problems (it is the only one I have with 16k ram; put code into a 27128). Did a multi-connect test and all seemed OK.

With 1.1.1f I again got into a "black hole" situation. In this case I apparently had FRACK set too short and TNC's async buffer over-ran without sending an X-OFF. After the buffer flushed itself on the air, data from computer (terminal) to TNC was apparently getting to TNC and going out, but there was no echo from TNC and no data from TNC to terminal. It almost looks like TX buffer is overwriting some crucial part of RX code. The only way to clear the fault was to power down. In our user notes we MUST specify that FR must be big enuff to take the outgoing packet; rule of thumb I used is that FR should be numericaly > or at least = MAX. I have had this "black hole" report from a couple of local users too. Ideas?

73, Tom

C2987 CC206 Lyle Johnson (WA7GXD,2973) 10/16/85 2:09 AM L:26
KEYS:/BEACON BUG?/INFO NEEDED ASAP/ONWARD!/

Howie,

One of the local beta testers, Eric, N7CL, notes the following on 1.1.0 code.

Beacon After N seems to reset itself after it sends the beacon and thus turns into s beacon every N. Highly undesireable. I also must note that Eric runs with no squelch so the DCD line will have some chatter on it. I may have missed an earlier comment that deals with this.

Also, we are about to print up the errata sheets/software notes for the next (last?) batch of TNC 2s. Any details you can give me on any of the commands not found in 1.1.0 will be very useful -- especially the treatrment of streamswitch when users = 1 and users <> 1 and the impact on transparent mode. YUou have seen the earlier systems manual, so you know the kind of info I need. We hope to send out a hundred or two kits by early next week with 1.1.1 EPROMs (I spent tonite writing a pascal filter to take the offset files in the 1.1.1 hex dump and offset them back to the correct address -- the eprom programmer we bought doesn't support an offswet address!!!! I needed the experience, it workls fine noew, I had nothing else to do anyway...)

Cheers!

Lyle

M 4082 Dave Pedersen (N7BHC,2960) 10/16/85 9:07 AM L:21
KEYS:/TNC-2 PROBLEM?/
TO: (Group 95)

One of our group here in Salt Lake has discovered what may be a bug in the TNC-2. I have duplicated the problem on my TNC-2. Is it a problem on the TNC, or just pilot-error?

FROM: Steve Baxter KA7JXR
RE: TNC-2 Problem ?

A few days ago, I decided to experiment with the 8bitconv command on the TAPR TNC-2. When I turned it ON, the TNC appeared to develop problems; it would not respond to any command except control-X, thus I could not get out of the 8bitconv mode, or do anything else either. The only solution I found was to unplug the RAM chip.

Is this a problem on the TNC-2, or am I just not doing something right?

73...Steve KA7JXR

M 4255 Howard Goldstein (N2WX,2987) 10/16/85 6:18 PM L:11
KEYS:/TNC-2 PROBLEM FIX/
A: 4082 TO: N7BHC, (Group 95)

Dave,Steve,

I'm sorry confirm what you discovered already; 8 bits were also being looked at by the TNC in cmd: mode, belying the CONV part of 8BITCONV. This will be corrected in the next distribution release!

If you want to run 8BITCONV on with your release, make sure AWLEN is set to 8 and your terminal/computer is set up to handle 8 bits. Provided commands and parameters are issued with the 7th bit low, it should work okay until the new s/w comes out.

Thanks for the report!

73 Howie

(Modification:)
C2987 CC200 Howard Goldstein (N2WX,2987) 10/16/85 6:04 PM L:0
(ORIG.) 10/14/85 7:47 PM L:38
KEYS:/WAS 1.1.1(G)/UPDATED IN CC207/

C2987 CC207 Howard Goldstein (N2WX,2987) 10/16/85 6:03 PM L:58
KEYS:/HERE WE GO AGAIN/+READ FOR 1.1.1(H)/CKSUM $7F/REQUIRES 16K RAM/

To: Beta
Fm: Howie
Re: 16K RAM version 1.1.1h, ten connection, release description
Dist: Closed, for now

The .HEX of the release described hr can be had with +READ @(thisitem).

Release 1.1.1(h) - checksum $7f - follows on the heels of (g) pretty quickly and fixes (again) something, changes (again) something, and adds (again) a few things:

o - STREAMCa ON|OFF   default: OFF. When ON, the callsign (if connected) of the other TNC is displayed between colons immediately after a TNC originated stream switch. Ex:

```
 |A:K4NTA-2:hi howie
 hello ted how goes it?
 |B:WA7GXD:*** CONNECTED to WA7GXD
 |Bmust be a dx record. ge lyle
 |Aunreal ted! fl-az no digis!
 |B:WA7GXD:big band opening...ge
       etc.
```

What used to be like "|A" is now like "|A:KV7B:" if the parameter is ON.

o - STREAMDbl ON|OFF   default OFF - This setting, when ON, will send all STREAMSwitch characters, except those that result from real STREAMSwitches originated by your TNC. STREAMSw must NOT be one of the physical stream letters used by your TNC for this to operate correctly.

o - ** DELETED - TRAPSTREam ON|OFF (replaced by STREAMDbl)

o - ** FIXED - 8BITCONV affecting commands problem

o - ** FIXED - version 2.0 protocol problems

o - ** FIXED - Errant Beacon AFTER startup problem

----- Below are earlier 1.1.1(g - checksum $2f) changes

TNC 2 software for 16K RAM is done. Buffer enlargement is one of the things addressed. Here are some other changes:

o - Up to TEN connections/links are allowed

o - A clock adjustment is installed. The associated command is CLKadj nnn, where 0=<nn=<65535. The adjustment effects the time by subtracting 9.1667ms every 100nn msec. Ex:
only WHEN CLKadj != 0,

$$\text{speed change of clock, in \%} = 100 - 9.16667 * \frac{1}{\text{CLKadj}}$$

o - *** DISCONNECT messages are stamped according to CONstamp

o - USER command is changed to USERS, and

o - The USERS parameter has new significance:

```
     USERS = 0  allows incoming connections on any free stream
     USERS = 1  allows incoming connections on stream A only
     USERS = 2    "   "      "   "     "       "   on streams A & B
     USERS = 3    "   "      "   "     "       "   on streams A,B, & C
           etc...
```

o - A bug where the output stream didn't change on ^C from CONV mode on a different stream has been fixed.

73 Howie

M 4486 Thomas A. Moulton (W2VY,995) 10/17/85 11:59 AM L:8
KEYS:/TAPR UPDATE/
TO: (Group 95)

Here is your TNC 2 Update for Wednesday October 16

The cabinet end panels have been mailed for up to order number 511

The TNC 2's will be shipped starting the week of Oct 28
This will be starting with order number 512

Orders are still being taken for the November time frame

C2987 CC208 Howard Goldstein (N2WX,2987) 10/17/85 4:37 PM L:94
KEYS:/SOFTWARE NOTES/COVERS 1.1.1 CKSUM $7F RELEASE THROUGH 1.1.0/
A: 207

o - The signon checksum is $7F

o - STREAMCa ON|OFF default: OFF. When ON, the callsign (if connected) of the other TNC is displayed between colons immediately after a TNC originated stream switch. Ex:

```
 |A:K4NTA-2:hi howie
 hello ted how goes it?
 |B:WA7GXD:*** CONNECTED to WA7GXD
 |Bmust be a dx record. ge lyle
 |Aunreal ted! fl-az no digis!
 |B:WA7GXD:big band opening...ge
       etc.
```

What used to be like "|A" is now like "|A:KV7B:" if the parameter is ON.

o - STREAMDbl ON|OFF default OFF - This setting, when ON, will send all STREAMSwitch characters, except those that result from real STREAMSwitches originated by your TNC. STREAMSw must NOT be one of the physical stream letters used by your TNC for this to operate correctly.

o - CLKADJ nnnn default 0 - zero is no adjustment only WHEN CLKadj != 0,

$$\text{relative clock spd.} \quad , \text{in } \% = 100 - 9.16667 * \frac{1}{\text{CLKadj}}$$

o - USers n default 1 - The parameter may take on values from zero through 10. USers affects ONLY how incoming connect requests are handles, and has no affect on the number or handling of connections you may initiate on your own.

```
          USERS = 0  allows incoming connections on any free stream
          USERS = 1  allows incoming connections on stream A only
          USERS = 2    "  "    "  "    "      "   on streams A & B
          USERS = 3    "  "    "  "    "      "   on streams A,B, & C
               etc...
```

o - TRIes n 0<=n<=15 default ? - This command is used to retrieve and set the count of tries on the currently selected input stream.

If RETRY is set to 0, the value returned by issuing a TRIES command will always be 0. Use of this command to force a new count of tries is NOT recommended.

o - STReamsw n 0<=n<=255 default $7C "|" - The STReamsw command is used to select the character used by both the TNC and the user to indicate that a new stream - that is, a connection - is being addressed.

The character can be PASSed in conv mode. It is ALWAYS ignored as a user-initiated stream switch in transparant mode, and flows through as data. This means that the outgoing stream can not be changed while "on line" in transparant mode. <<see further STREAMDb and STREAMCa commands>>

o - CStatus - This is an immediate command which shows the stream identifier and link state of all ten streams (links), the current input and output streams, and whether or not each stream is "Permanant" (see CONPerm). Ex:

```
cmd:CS
A stream - IO Link state is: CONNECTED to 305MLB
B stream -    Link state is: CONNECTED to AD7I P
C stream -    Link state is: DISCONNECTED
D stream -    Link state is: CONNECTED to WA7GXD via K9NG-2
          ...
I stream -    Link state is: CONNECT in progress
J stream -    Link state is: CONNECTED to KV7D via KV7B-1
```

The example above shows the A stream is assigned both the input and output streams and the B stream is connected to AD7I "Permanantly". All the other streams' states are shown as they'd normally appear when a CONNECT command without parameters was issued.

o - RESTART - This is an immediate command which has the same effect on the TNC as shutting it off and turning it back on again.

o - MYAlias ccccc Default: blank - The parameter of this command is an amateur callsign-ssid that this TNC recognizes for digipeating purposes ONLY.

In some areas, wide coverage digipeater operators have changed the callsign of their machine to a shorter and (usually) easier to remember identifiers like ICAO airport id's or other well known abbreviations.

Use of this command permits HID to identify normally with the MYCALL callsign yet permit an alternate "alias" repeat-only call.

-*-*-*-*-*-*

There are a few operational changes in this release that should be noted:

- All connect requests from stations with totally blank callsigns are rejected with a busy response.

- The *** DISCONNECT prompt is daytime stamped according to the CONSTamp setting

- TNC 1-like transparant mode is supported only when one connection is established and USERS is set to 1

-*-*-*-*-*-*

- in the TNC 2 System Manual, chapter 6 p 37 = MHeard command, change the line that reads:
Stations that are heard digipeating are marked with a * in the heard log.
to
Stations that are heard through digipeaters are marked with a * in the heard log.

C2987 CC209 Tom Clark (W3IWI,2976) 10/17/85 10:06 PM L:13
KEYS:/HOWIE: "BLACK HOLE" BUG?/LYLE: HEX => BIN CONVERSION FOR PC/

Howie: were you able to do anything about the "black hole" bug where the TNC2 refuses to echo anything when it gets an input buffer overflow with FRACK too short to support the outgoing packets? I've now had 6 local users report this one to me. My standard fix is to tell them to increase FRACK and to pull the RAM to achieve a reset from the bogey condition.

Lyle: I have a PC program that converts Intel HEX into Binary. It is written in BASICA and runs quite fast compiled under BASCOM. It does handle offsets and also insures that all trailing (unused) bytes in ROM are defaulted to $FF. Need a copy?

73, Tom

C2987 CC210 Howard Goldstein (N2WX,2987) 10/17/85 11:06 PM L:11
KEYS:/BLACK HOLE/
A: 209

Tom -

I'm going bonkers trying to reproduce this one - really!

I set up for AW 8 PAR 0, leave XFLOW/XON/XOFF at default, set MAXFRAME 7 and FRACK 1. Then connect via 2 digis and open er up but still no crash. Note my data source ignores flow control from the TNC for this test and keeps sending the whole file, overflowing every char of the way.

Other commands left at defaults.

Will try gorging it interfaced a few different ways tomorrow and see what happens.

73 Howie

M 4853 Dave Pedersen (N7BHC,2960) 10/18/85 8:51 AM L:12
KEYS:/56KB MODEM/DIGITIAL AUDIO/
TO: (Group 95)

Bill Gillman, WA7RFS, in Salt Lake City, is interested in experimenting with digital audio transmission, and poses the following questions:

Is anyone else playing around in this field?
What developmental work is being done with a 56kb modem?
Has anyone used the TNCs at 56kb?
Where can I find information on the modem used in Swedish Softnet system?

I hope someone can be of help to Bill.

73...Dave N7BHC

M 5202 Steve Goode (K9NG,2981) 10/19/85 12:11 PM L:19
KEYS:/56KB MODEM?/
A: 4853   TO: (Group 95), K9NG

Anyone responding to Dave Pedersen's request for info on the Swedish Softnet system please send a copy to me also. I have heard rumors of this network for about a year but have never received any solid info on it. I have received requests from Germany on my 9600 bps modem and wonder why they would be interested in such a slow date rate (compared to 56 kbps) if the 56 kbps is really available. Any solid info would be greatly appreciated here.

I know of there groups that are interested in higher data rates than 9600. They all have my modems going. The problem is that the people interested in higher rates are also the people installing the 9600 bps backbones (as I see it) so until the first backbones are in place and these people are free for new experiments it will be difficult to get higher rates.

At this time any higher rates would probably be scaled versions of the 9600 bps modem requiring wider receiver IFs. Anyone with other ideas please send them along.
73
Steve

M 5300 Pete Eaton (WB9FLW,2970) 10/20/85 12:55 AM L:38
KEYS:/TNC 2 BLUES/BUT THE CARDS WILL WIN THE SERIES/
TO: (Group 95), WB9FLW

To: All
From: Pete WB9FLW
Subject: How NOT to put your TNC 2 rev 2 board together!
Dist: Open (but...NO smart a-- replies allowed!)

This evening Mel Whitten K0PFX and I were finishing off my TNC 2 rev 2 board, this revised circuit board requires a small piece of coax that acts as a faraday shield from the dc power plug to the 7805 regular. While doing this simple proceedure we were watching the St. Louis Cards destroy the K.C. Blues.

Since a child of 10 could solder this coax to the board we spent our mental energy on the game. When we turn the TNC 2 on to smoke test it the results were positive, it did smoke. Seems that for some reason we had bypassed the 7805 5 volt regulator and applied plus 15 volts to the TNC 2's 5 volt line.

This does not work.

Nor does the TNC 2 work, at all. Now all the lights on the front light....all the time (makes a great night light) but besides that nothing.

I hope that TAPR places in their Erratta sheet "do not watch the world series and build this kit" or "please allow your 10 year old to construct this unit if you must watch sports programs at the same time."

Well now that I have alot more free time I can watch the series in the knowledge that my hobby will not interfere...at all.
P.S. the series will not go to 7 games the cards will stomp the Blues BEFORE game six.

P.P.S. don't pay any attention to someones predictions who can put a silly TNC kit together.
PLAY BALL!

C2973 CC4 Jack Brindle (WA4FIB,2963) 10/18/85 7:51 PM L:80
KEYS:/NETWORK DATAGRAMS VCS PVCS/

Phil; But what is your mailing address???

One of the things that really bothers me about the datagram approach is the necessity to contain routing information in every packet, and the resulting length of the packet. I agree that at present we have datagrams in layer 2, and that someday we should get rid of that. (prrobably it will happen about the time that the PID becomes useful - no present TNC code that I have seen allows access to the PID!) The biggest reason to get rid of the relay fields in L2 is that it just doesn't belong there. It served a purpose, but L3 is supposed to eliminate that. Yet it may continue to serve a purpose once L3 is installed.

The name of the game in radio packeting is to maintain short transmissions so that the transmitted data does not get "clobbered" by QRN and fading. The implimentations of datagrams that I have seen assume a very good transmission media, and place extra info in the routing header to lengthen the packet. Of course when one is transferring 4096 byte packets over coax, who cares. Rather than hit you with more stuff that I'm sure you have already thought of, it's probably best to let you explains things first. As soon as I get the TCP/IP info from you (your address, please...) I'll have a better shot at asking intelligent questions.

As a networking person, I too have some ideas on how to impliment the sye system. Feel free to blow these away...

I have problems with the overhead caused by the virtual call. In a limited situation, it is probably ok, but we will fill the airwaves with a lot of packets just establishing the layer 2,4, and 5 connections, not to mention lots of L3 requests inside the network. In a limited fashion, virtual calls do have their place. Lets assume a virtual circuit system. The VVirtual call should then be used to make connections with nodes that for some reason are not known to the network. This also suggests that some mechanism must exist to alleviate the VC. The answer lies in between the virtual call and the datagram. It is, of course, the Permanent Virtual Circuit. The PVC uses established routing paths well known to all nodes, so that circuits do not have to be built, simply used. The biggest problem with PVCs is the necessity to maintain routing tables in each node. This would cause a "Control" node to be designed into the system to oversee the network and build new tables for each node as new nodes appear. It allows a close-knit system that can be monitored for problems. It also would allow network security to assure that unwanted stations do not invade the sanctity of the system by attaching themselves where they do not belong.

This brings up another question, though. Just where does the network and transport layer code belong? I believe that some form of transport code belongs in the TNC, since the end TNC's are the only communications entities that can verify data on an end-to-end basis. However, we should take a look at making the network look like a black box to the end users, much like telenet does for this connection. The advantages become numerous since the TNC can remain simple, like the Bell 103 & 212 modems we use with telenet (although those have gotten awfully complex). All information necessary to packet routing then is placed in the network nodes, and the end user's TNC can continue to think he is in a point to point connection. The biggest advantage is that the current TNCs can be used with very little modification. This is a major consideration when you look at the various TNCs on the market and the fact that few of them adhere to the current AX.25 standard. This is slowly being fixed as the TNC2 comes out, along with TAPR 4.0 or the WA8DED code for the TNC1. Our biggest problem, then will be the "popular" Kantronics unit, which is definitely non-standard with no hopes in the future of becoming standard (in fact, just the opposite - apparently they are trying to create their own "standard"). It will be a tremendous task to upgrade all these TNCs to use the "new" layer 3 code, whatever it is. Let's face it, packet has been taken away from those of us who like to experiment and build, and given to those who like to buy and "operate."

These are definitely thoughts to consider. I don't necessarily agree with a lot of them, being a network "purist" I'd like to see a lot of thinds done differently, but like the repeater fields, they are here and we might have to accept them.

It seems that PVCs have not yet been proposed to packet radio in general. Please give them some thought, then tell us why you think they will or won't work. Implimented properly, they could give the best of both worlds, VCs and datagrams, at an advantageous cost! One thing I would see in them is that regional networks would be necessary because of the large job of building nationwide routing tables! Oh there is another advantage to having a central controller. It would make software distribution and upgrades easier. Each node would be outfitted with a "baseline" program that would allow the node to boot itself into the network. Then the central node could maintain the latest code release in each node, simply by downloading the new code to the node. No more mailing disks & EPROMS!

Anyway, much to consider. My Session layer protocol is moving along also. I expect to release it at the Atlanta Packetfest. It is extremely interesting and worth a peek. I even hope to have it running in MacPacket by then!

More later. 73, Jack B.

C2973 CC5 Phil R. Karn (ka9q,2979) 10/19/85 2:57 AM L:237
KEYS:/NETWORKING MODELS/

Jack, thanks for your interesting comments.

The overhead of a protocol is a function of both the size of its header and the amount of data the user sends in each packet. More specifically:

efficiency = data/(header+data).

If the header size is increased, it is possible to retain the same overall efficiency by sending more user data in each packet. It is instructive to compare the real numbers used in some actual networks, namely Telenet (X.25) and the ARPA Internet (TCP/IP).

X.25 levels two and three have a combined header of 9 bytes (the 6 bytes in layer 2 includes the closing flag and CRC). The maximum packet size on Telenet is 128 bytes, and we haven't yet figured out a way to negotiate it to a higher value. TCP/IP has a combined header of 40 bytes, but every Internet host is prepared to receive datagrams of at least 576 bytes (512 bytes data plus TCP/IP plus room for options.) Let's compute efficiencies, assuming maximum packet sizes as would be the case during a file transfer:

Telenet/X.25: 128/(128+9) = 93.4%
Internet/IP/TCP: 512/(512+40) = 92.7%

Hardly much different. In fact, many hosts, including ours, have TCPs which negotiate the maximum packet size limit up to 1K, since memory is cheap:

1024/(1024+40) = 96.2%

This is better than Telenet! One almost wonders how serious some of the X.25 advocates really are about header overhead when you look at some real networks. Of course, this is a slightly unfair comparison, since TCP provides end-to-end guarantees and X.25 doesn't. We should add to the Telenet case at least the overhead of a "lean and mean" transport protocol designed for operation over a virtual circuit subnet (e.g., TP-1), and this can only make the Internet look better.

A valid concern, of course, is that the user may not generate large enough packets to overcome the handicap of the larger datagram headers. This was a very real problem with early implementations of TCP when used for remote login over very slow network paths. The Internet presentation-level protocol for remote login, "Telnet", (not to be confused with TelEnet) is often "negotiated" into character-at-a-time mode in order to simplify the use of screen editors. The sender's TCP, since it usually faces a generous window of 1-2K bytes on the receiving machine, would gladly generate a 1-byte TCP segment (TCP packets are called "segments") each time the user hit a key, even if the network was already clogged with lots of little segments still in flight. This became known as the "small packet problem".

A solution for the small packet problem was found by John Nagle at Ford Aerospace. It is elegant and effective enough to now be universally implemented. Basically, it imposes a "one segment at a time" restriction on the sender. If the user hits a key when the connection is idle, a single character segment is sent. However, any further characters are held in a queue at the sender until an acknowledgement is received for that character. When the acknowledgement does arrive, the pending characters are sent out in one large segment, and the process repeats. Only when there is already enough data pending in the transmission queue to generate a maximum-sized segment will more than one segment be allowed to exist in the network at one time.

A little thought will show that this scheme is highly effective at reducing the number of small packets in the network and the possibility of network congestion, and does so with little effect on user-perceived performance. It is a superior replacement for the packet closeout timers often found in X.25 pads, and is easier to implement, too. Still, it is clear that the best solution for interactive users is to find a way to avoid altogether the need for character-at-a-time transmission, but when one is talking to an application that doesn't always operate in terms of "lines", this isn't easy.

I think the trend will be towards smart "network terminals" that understand something about the specific application and can offload many of the functions now done on the hosts. When this is done, "terminal to "host" traffic will look more and more like "host to host" communication: fewer but larger packets, and a need for more flexible networking services such as those provided in the Internet. This is perhaps THE sharpest contrast between the CCITT and ARPA "universes". The 1970's model of a "dumb terminal" talking through a PAD to a big IBM mainframe thoroughly permeates the design of X.25, and since this is exactly what 99% of X.25 users still do, it's no real surprise that it's optimized for this task. But it's hardly a general purpose packet network, and the growth of personal workstations and local area networks (which are invariably based on datagrams, e.g., Ethernet and PROnet) are already forcing the public data net people to reconsider the services they provide.

You make a valid point about the increased noise vulnerability of large packets. To a certain extent I think that we will be forced to improve the BER of our links anyway just to get reasonable performance, but that begs your question. There are two ways that large datagrams might be broken down into smaller noise targets. One can "internally fragment" each datagram as it is sent over a Level 2 link. For example, AX.25 might forward a single 1K-byte IP datagram as a sequence of four 256-byte I-frames, and the receiving layer 2 would put these back together before passing the datagram back up to the layer 3 module for further processing. Clearly only the first I-frame need carry the IP datagram header, but the link layer must provide sequenced delivery of each piece of the datagram (AX.25 does) with some indication of the beginning and end of each multi-frame datagram (AX.25 doesn't, because this is the function of the M-bit in the Packet Layer of X.25, and AX.25 doesn't have this layer.)

So that it can run on a pure datagram network (e.g., Ethernet, or AX.25 in the disconnected mode with UI frames) IP provides another option, namely "external fragmentation", where a datagram can be broken up at the IP level into IP fragments. They propagate through the Internetwork as though they were separate datagrams and are reassembled only when the reach their final destination. Of course, this duplicates the IP header on each packet and increases overhead, especially since other nodes see extra packet loading as a result. In practice, IP-level fragmentation is reasonable as long as the fragments aren't too small, while link-level fragmentation is best when one link has an unusually small packet size limit. Both methods can and do coexist on the Internet.

You mentioned at the beginning of your note that datagrams "contain routing information in every packet". This is not necessarily so. At a minimum, they must carry only the source and destination addresses in each packet; in IP, addresses are 4 bytes so this is 8 bytes of overhead (included in the 20 byte IP header). Even AX.25 without digipeaters will still require source and destination callsigns, so it remains a true datagram protocol at the lowest level. (In fact, I challenge anyone to give me an example of a protocol operating on shared, multiaccess media that ISN'T datagram-oriented at its lowest level.) Providing routing info in each packet is called "source routing", and AX.25 with digipeaters is an example of this. IP also allows source routing, but unlike AX.25 it is OPTIONAL, and it is not necessary (or even common) that source routes appear in each datagram. Ordinarily, routing tables at each packet switch are consulted to determine the best "next hop" toward the datagram's destination; source routing allows senders to override the routing tables when desired (e.g., to perform network testing or to bypass a routing problem.)

You make some interesting comments about permanent virtual circuits. It has occurred to me that datagrams and switched virtual circuits are merely two points on a much larger, continuous spectrum of network design:

```
Fully                                              Fully
Static                                             Dynamic
Bandwidth                                          Bandwidth
Allocation                                         Allocation

  |----------------|---------------|----------------|
Dedicated       Circuit         Virtual          Datagrams
Lines           Switching       Circuits

<------ Increasing initial connectivity/setup delay
        Increasing per-transmission overhead ------>

        Increasing suitability for bursty traffic-->
<------ Increasing suitability for predictable, steady traffic
```

Now what do you mean by a "permanent virtual circuit" (PVC), and where would it lie along this line? Does an implicit PVC always exist between every pair of nodes in a network? If so, the "PVC identifier" has to be just as large as the address field. This negates the only selling point of a VC protocol, namely not having to send big addresses in each packet, and your PVC is really the same as a datagram service (except for side issues like guaranteed delivery and sequencing, which may or may not be associated with any particular datagram network).

On the other hand, if PVCs exist only among a subset of nodes, i.e., like a limited collection of dedicated lines, how does a pair of nodes without a direct PVC communicate? Either there has to be a simple, automatic mechanism to create a PVC (which means that they're no longer "permanent"), or there needs to be ANOTHER layer of routing and switching on top of the PVCs in order to provide full connectivity to the higher layers. So I don't really see how PVCs are relevant to amateur packet radio. They are a special service provided to commercial entities that aren't interested in communicating with more than a few well-specified places, but don't have the traffic and/or money to justify leased lines.

Regarding the placement of code: this should be fairly straightforward from the ISO and ARPA reference models. The transport layer must be end-to-end, otherwise it cannot guarantee end-to-end data integrity (which is its major function). In other words, it resides in the client host (TNC or whatever) at each end of a logical "connection", but not in the intermediate packet switches ("transit nodes"). The host must have enough understanding of the network layer in order to exchange traffic with the nearest packet switch, but it need not concern itself with switching and routing (unless it also serves as a packet switch, of course.) This is absolutely trivial with IP, because "speaking IP" requires only that you stick the appropriate header in front of each TCP (or other upper level protocol) packet and send it at your nearby packet switch. The packet switches need not concern themselves with TCP (unless they are also capable of being hosts), but instead must be able to do the things that packet switches do: handle multiple links and route packets between them.

Some people have proposed a networking model that looks like this:

```
Unmodified -(AX.25)--Magic Link---(backbone net----Magic Link -(AX.25)--Unmod
AX.25 TNC            Box           protocol)        Box                  TNC
```

The "Magic Link Box" must now act as a "protocol conversion gateway" (a.k.a. a "transport level bridge"), splicing an AX.25 connection on one side to the "backbone net protocol" (e.g., TCP) on the other. Such Magic Link Boxes have been drawn many times on viewgraphs and on slick ad copy, and at first they look really nice because nobody has to change their existing equipment.

---> HOWEVER, THIS APPROACH IS SIMPLY WRONG!!!! <-------

It is only when somebody actually tries to build or use such a box that they realize how flawed the model is. The first (of many) problem(s) is that there is no longer any end-to-end protocol; TCP can only guarantee delivery between the Magic Link Boxes, not between the TNCs on each end. Other problems: How does the user tell the Box who he wants to connect to? How does the Box tell whether what's coming in on the AX.25 link is to be relayed as data or acted on as a command? Suppose the user wants to maintain multiple connections; how does he indicate multiplexing of his data streams? Suppose the Magic Link Box he's connected to gets hit by lightning; how can he switch over to another Box in his area?

And so on. By the time you've solved these problems, you finally realize that you've invented a full-blown but utterly ad-hoc "protocol" for commanding the Box over the AX.25 link. Whether you've added this "protocol" to the TNCs (so they are no longer "unmodified") or forced the user and/or his host system to learn it (more likely), you will have discovered that this was far more painful and kludgey than it would have been to do it right in the first place by putting the transport protocol in the TNC or other host where it belongs. The Boxes can then just switch packets, and that's certainly enough to keep them from getting bored (especially Z-80s, since it has been arbitrarily decreed that we shall use them here!)

I put Protocol Conversion Gateways in the same class as Star Wars. Both are superficially alluring. On closer examination, however, they turn out to be extremely difficult, expensive, dangerous and ultimately futile attempts to devise "technological fixes" for what are fundamentally political problems. TANSTAAFL!

One last subject: the session layer. I've never been able to get a clear picture of just what it is that the session layer does, and perhaps this is because its function seems to be unnecessary in any network save SNA. For example, TCP has source and destination port multiplexing, so that one TCP module in a host can handle all of the network traffic from a number of user processes. Depending on your point of view, this means that TCP combines levels 4 and 5 into a single layer, or it makes level 5 unnecessary.

Sorry this has gone on so long, but once you get me going...

73, Phil

C2987 CC211 Lyle Johnson (WA7GXD,2973) 10/19/85 4:25 PM L:15

Howie,

A couple of ?bugs may be in the 1.1.1 release.
1) ctrl-Y (canpac) doesn't work to cancel dispaly output as documented in the system manual chap 6 page 14. I think you need to ctrl-s the data flow, then ctrl-y works...
2) whewn using a 2-wire rs232 hokup (+ gnd) the tnc 2 refuses to send data to the user port unless the CON led is illuminated. Sign on message terminates when the two CON and STA leds extinguish, altholught e TNC accepts date, it won't echo it until the CON LED is again illumkinated...
3) if you are connected to a station, then force a disconnect by enetering the DISC command twice in rapid succession, the CON led operates backwards 9off when connected, on when disconnected).

Lyle

C2987 CC212 Howard Goldstein (N2WX,2987) 10/19/85 7:50 PM L:10
KEYS:/LEFT OUT EARLIER COMPLEAT GUIDE OF 1.1.1 COMMANDS/RECONNECT/BUGS/

Lyle - will check out bugs in prev comment.

I left out a command from the earlier "compleat" guide of new commands:

REConnec call1 [Via call2[,...]]

The parameter call1 MUST match the callsign of the station already connected to on the current input stream. Other parms as per required for path.

It's illegal to either: issue this command while link state is: DISONNECTED (and a '?not while disconnected' message appears) or change the call1 parameter from the current station.

73 Howie

C2987 CC213 Skip Hansen (WB6YMH,2964) 10/19/85 8:08 PM L:28
KEYS:/HARDWARE FLOW CONTROL ?/

Howie,

Got another one for you. I was sending a file to another PBBS today using hardware flow control (stop=start=xon=xoff=0, rflow=xflow=off awlen=8,parity=0,autolf=on,8bitconv=off,nulls=0) in converse mode and data was getting dropped badly. I thought it was the PBBS until I tested with Harold and found that with exactly the same configuration in place the file would transfer perfectly in transparent mode, but drop large hunks (packets ?) when in converse mode. We tried with echo on and off, with no difference. The hardware flow control is going active and my computer is stopping, the echo from the TNC is perfect on the screen. The dropped data doesn't not seem to occur anywhere near where the TNC flow controlled the incomming stream. I also tried maxframe=1 and paclen=60 with the same results. The only way I was able to get the data thru converse mode was by sending 16 characters then waiting for about 1 second, then sending another 16, etc. The channel was very uncongested (4:15 am) at the time of the test. I also tried direct as well as thru 2 digipeats with no difference.

I just ordered the 8K X 8's and 27128's to upgrade Tiny with, but I may end up stealing the RAM from my TNC-1 if I get impatient.

In case you are woundering why all of the activity from me of late I have been on vacation for 2 1/2 weeks, but fear not, I am going back to work next week! Thanks for the fast response on my stream switch suggestion!

73's Skip WB6YMH

C2987 CC214 Tom Clark (W3IWI,2976) 10/20/85 12:56 AM L:44
KEYS:/WORLI BBS COMPATIBILITY/ALSO: VERSION 2 COMPATIBILITY TEST RESULTS/

Howie, there have been several EASTNET msgs going back and forth between WORLI and me about handshaking/over-run problems with TNC2. The main problem now seems to be in the monitor mode, with some characters being dropped. Apparently Hank handles the handshaking with the TNC2 in this mode using FLOW -- when he wants the TNC to go mute he sends a <space> and to get it to resume he sends a <bs>. Apparently this works fine with the TNC1, but TNC2 usually gets a couple of extra characters in there so that the <bs> doesn't obliterate the <sp>, but rather some other characters. I manually simulated this on the BBS 820 in terminal emulator mode. If XF ON and if xon/xoff characters are used, the problem seems to go away, so I have suggested to Hank that he try more conventional xon/xoff handshaking. More details later after I get an answer from Hank.

I have now received RLI 10.2 BBS which now recognizes all the new monitor mode outputs and properly logs activity under the "J" command. Let me know if you want a clone.

All: We ran an interesting AX25 Version 2 compatibility test here this weekend. I put the BBS on with AX ON to see if we would have any problems. The news is good.

As was expected, Version 1 users connecting to the BBS caused the TNC2 to revert to Version 1 protocols, so the change was transparent to them. Those users running TNC2's or WA8DED TNC1's with Version 2 found that everything worked fine for them too, except that (as we knew would happen) those who used TNC1 Rev.3.x digipeaters or Kantronics digipeaters couldn't get their packets thru them. In this area, most of our major digi's are using old VADGC boards with AJ9X code and they handle Version 2 packets just fine.

Since the BBS runs as an answerer most of the time, the user was in control of which version was in use. Since the BBS forwards mail to other BBS's, and since that forwarding involves digipeaters, it was necessary to put an AX OFF at the start of the FWD.TNC forwarding control file, and restore operation to normal by including an AX ON at the end of the file.

Based on these tests, I have decided to leave the 145.01 input port on W3IWI BBS set with AX ON as a positive step towards future network improvements.

73, Tom

M 5850 Tom Clark (W3IWI,2976) 10/21/85 11:22 PM L:32
KEYS:/A LITTLE FUN/OUR NETWORKS DO WORK!/4 BBS'S LINKED/
TO: (Group 95)

Thought you might find the following message fun -- as a test of Gateway operation tonite I tried a quadruple bounce with the connection described in the text of the message, ultimately reaching N2WX in Florida. I sent a message to Howie and then read it back, and here was the transaction:

```
                                    - - - - - - - - - - - -
W3IWI de N2WX: Last msg # 2432
(B,D,G,H,I,J,K,L,R,S,T,U,W,X,?) >
r 2432
Msg# TR  Size  To     From    @ BBS  Date     Title
2432  N   764 N2WX   W3IWI           851021 Heights of futility!
```

Howie, this probably represents the ultimate packet radio folly.

I am operating TNC2 (yep, with 1.1.1h!) on 145.01 as W3IWI-1 connected to W3IWI. W3IWI BBS is operating as a Gateway to 145.05, connected to K3VPZ
K3VPZ is operating as a Gateway to 14.103, connected to K4NTA
K4NTA is operating as a Gateway to 2 meters, connected to you via STU,MLB
and finally I am logged in on ur BBS sending this message!

Packet radio and our networks do work! And all in all, it is about as fast as EIES is sometimes.

1.1.1h seems to work fine. I like RECON. Haven't explored all the ramifications yet. More feedback later. Good to chat earlier in the day. Good luck on getting the software defined. Posted the SAREX2.004 report tonite on 2978 as an update following a busy day today

73, Tom

```
W3IWI de N2WX: Last msg # 2432
(B,D,G,H,I,J,K,L,R,S,T,U,W,X,?) >
```

M 6144 Thomas A. Moulton (W2VY,995) 10/22/85 4:17 PM L:12
KEYS:/AN UNPLEASANT DUTY/BUT WE HAVE A NEED FOR SOME NEW PEOPLE ON FOR A NEW PROJECT IN DRNET/
TO: (Group 95)

I am catching up in my G95 duties...

We have been working on many things,
and one of the projects is going to require a few new accounts,
it is very important and I don't want to have to get out my axe,

I would like to see if there are any users that feel that
their account is not being used to it's full potential and volenteer it.

Please send me a message if you would like to help out.
(if this doesn't work I may have to ask for a justification statement from
each of you... I really don't want to wade through all the mail tho...)

M 6319 Jack Brindle (WA4FIB,2963) 10/23/85 12:21 AM L:16
KEYS:/OCTOBER DOES NOT MEAN NOVEMBER./
TO: (Group 95)

To: all MacPacket users awaiting TNC2term.
From: Jack Brindle, WA4FIB

We have decided to delay shipping MacPacket/TNC2term until at least December 1. The reason is the lack of hardware to test with. TAPR promises a TNC2 REAL SOON NOW... I am told that will be after November 1. I am trying to introduce the new version in unison with a major MacPacket upgrade, now scheduled for early December.

I understand TAPR's desire to ship the latest and greatest, but they should at least keep the folks they are holding orders for posted on their progress. Many TNC2 orderers are under the impression that TNC2s were to go out on the 15th. Anyway, I KNOW it is a volunteer organization and we must wait for the volunteers (we are all familiar with that, since packet radio has been developed by volunteers). In the meantime, those awaiting TNC2term, please have patience. Things can only move so fast.
73, Jack B.

M 6350 Lyle Johnson (WA7GXD,2973) 10/23/85   1:22 AM L:27
KEYS:/TNC 2 STATUS REPORT/
TO: WA7GXD, (Group 95)

TNC 2 update.

.

For those who have not called TAPR recently, the following is the TNC 2 situation.

.

TNC 2 through 511 shipped. We were shorted over 100 pc boards for our september delivery and, after careful consideration, changed PC houses. This has cost us about $2,000 in new setup charges and about $8.00 per board in purchasing costs.

.

The TNC 2 has been relayed out, the new ones have double memory and even better software.

.

Cabinet front and rear panels were sent out to TNC 2 Rev 1 owners.

.

By Thursday ofd this week (24 october) we will have everything for nearly 500 TNC 2s except cabinet front panels, due sometime next week.

.

We will ship TNC 2 Rev 2 without front panels, and send the front panels when they arrive. We should have up to order 900-950 shiupped out early the week of the 28th (we want to make it October if we can!!!).

.

Most units will be all CMOS. Extra RAM, extra EPROM< CMOS, new boards, now 5 LEDs (PTT added) all at no increase in cost. Thye manuals have been completely edited and reprinted as well.

.

Thank you for your patience. The wheels really are turning...

M 6353 Lyle Johnson (WA7GXD,2973) 10/23/85   1:28 AM L:1
KEYS:/PSR/
TO: (Group 95), WA7GXD

C2973 CC7 Phil R. Karn (ka9q,2979) 10/20/85   8:27 PM L:343
KEYS:/W3HCF TCP/IP PROGRESS REPORT/

>From @DCN7.ARPA:mills@dcn6.arpa Fri Oct 18 15:38:56 1985
Relay-Version: version B 2.10.2 9/18/84; site petrus.UUCP
Posting-Version: version B 2.10.1 6/24/83; site mit-eddie.UUCP
Path:
petrus!bellcore!decvax!ucbvax!ucdavis!lll-crg!seismo!harvard!think!mit-eddie!@DCN7.ARPA:mills@dcn6.arpa
From: @DCN7.ARPA:mills@dcn6.arpa
Newsgroups: net.ham-radio.packet
Subject: AX.25 and IP/TCP: a status report (absurdly long!)
Message-ID: <131@mit-eddie.UUCP>
Date: 18 Oct 85 19:38:56 GMT
Date-Received: 20 Oct 85 18:52:34 GMT
Sender: daemon@mit-eddi.UUCP
Organization: MIT, Cambridge, MA
Lines: 326

From: mills@dcn6.arpa
Folks,

Following is a longish report on current status and plans of my ongoing project to bring up a demonstration of the DoD Internet protocol suite operating at the network layer above AX.25. I first tried to hack the original TAPR-1 firmware to do this in both connection and connectionless modes, but gave up due to ornery problems in transparency and flow control. I next tried the WA8DED firmware and found some joy, pleasantries of which are reported below. The report is rather technical and intended for those with some protocol background and implementation experience, expecially in the areas of connectionless (datagram-oriented) networks and gateways.

Overview

I rebuilt what has become a monsterous driver for the LSI-11 "fuzzball" system, which is an old grizzly used for network protocol development, testing and performance evaluation and supports just about every protocol sung in the IP/TCP protocol suite. The LSI-11/73 I used for the present purpose (DCN6.ARPA) belongs to me personally and is located in my home along with 30 megabytes of disk, a swamp of radios, towers, a leased digital circuit to the office (hence the Internet) and related mud creatures.

The driver supports both the ordinary connection-mode, stream-ASCII operation commonly used to access BB stations, etc., as well as both connection and connectionless modes for Internet Protocol (IP) datagrams and its client Transmission Control Protocol (TCP), upon which application protocols like the TELNET virtual-terminal, FTP file-transfer and SMTP mail-transfer protocols are built. The station complement, consisting of TAPR-1, fuzzball and VHF transceiver, is organized as a full-function Internet gateway between subnet 128.4.1, representing the radio channel, and DCNET (128.4), which is itself gatewayed to the Internet via the ARPANET. It should be noted that no traffic capability between the radio channel and the ARPANET is expressed or implied in this document and that all traffic between the radio channel and any other nets reachable via nongovernment paths (Ford, U Michigan and U Maryland at present) is experimental and strictly noncommercial in nature.

The WA8DED Firmware

The WA8DED firmware ex box provides one connectionless-mode channel and five connection-mode channels, which can also be operated in connectionless mode. The TAPR is operated in host mode; that is, the fuzzball has to poll for everything, which it does at one-second intervals on all five channels. This handles the flow-control problem from the TAPR to the fuzzball. Since my TAPR is fully populated with memory (about 700 buffers) and TCP manages end-end flows anyway, I didn't bother with flow control from the fuzzball to the TAPR.

Host messages to the TAPR consist of the sequence

<channel number> <control code> <count> <text> ,

where <channel number> is an octet in the range 0-4, <control code> is an octet with value 0 for data and 1 for command, <count> is an octet representing the number of octets in <text> reduced by one and <text> is a string of up to 256 octets with no restriction on code combinations. Commands are in ASCII and are interpreted the same way as in ordinary (i.e. non-host) mode, except that the terminating <CR> is not included. No more than one command can be sent in a message and commands may not be split between messages. In the case of IP datagrams, the raw datagram itself is simply encapsulated in the <text> field as-is.

TAPR messages to the host have the same format as above, except that <control code> is an octet with values in the range 0-7, indicating success/failure, as well as status information, headers and data.

0normal response to command (no <count> or <text> field)
1normal response to command
2error response to command
3link status information
5monitor header
6monitor data
7user data

In all except codes 0,6-7 the <text> field is <NUL>-terminated. Codes 0-2 occur only in response to commands, while the remainder can occur spontaneously (but only in response to a poll, of course). Codes 5-6 apparently occur only on channel 0, while code 7 occurs only on channels 1-4. This creates problems in connectionless mode as described later. In the case of code 5 the <text> field consists of an internally formatted representation of the AX.25 header, which is followed immediately by the data (if present) as a code 6.

Upon initializatin the driver sends the sequence

<NAK><ESC>JHOST 1<CR> ,

which toggles the TAPR into host mode, if not already there, and is grumpily ignored if it is. The polling sequence, which is sent every second, consists of the five commands

<0><1><0>G <1><1><0>G <2><1><0>G <3><1><0>G <4><1><0>G ,

which empties the TAPR queues at a rate limited to one message per second on each channel. When lots of things are going on, such as when monitoring all channel traffic and managing a connection or two, the TAPR queues start to back up. The polling strategy and rate matching can obviously be improved.

Connection Management

The WA8DED firmware assigns incoming connection requests serially starting at channel 1 and continuing to channel 4 or a limit which can be set by a command. Completion of the handshake sequence results in a link status (code 3) reply on the assigned channel. Outgoing connection requests can be made on any channel as specified by the connect "C" command. In the case of channels 1-4, the request results in the AX.25 connection-setup procedure, while in the case of channel 0, the request specifies the route for subsequent connectionless (UI) frames.

If a data message is sent on channel 0 or on any other channel not preceded by a connect command, the data is sent as a UI frame. Connection data (I frames) appear as code-7 replies on the assigned channel and can be interleaved with data from other channels. Connectionless data (UI frames) can apparently be received only as monitoring data when enabled by the monitor "M" command and are preceded by a code-5 message indicating the route and protocol id. The protocol id can be changed by a command, but only on a global basis.

An AYT (ARPANETese for "are you there" aka "keep-alive") consisting of the RR supervisory message, is sent every two minutes on every active connection. If the max-retransmission counter is exceeded the TAPR attempts to reset the connection while holding on to the channel queues; however, if that fails the connection is aborted with an appropriate link status message. This can result in delays up to several minutes before the link-level protocol gives up and obviously will result in congestive collapse in retransmissions with conventional transport-level implementations.

The W3HCF Fuzzware

The TAPR driver constructed for the fuzzball can operate with both ASCII streams or IP datagrams in either connection or connectionless modes. The driver has an interface to the stream-oriented terminal I/O, so that local terminals and remote network (TELNET) terminals can connect via the driver and TAPR to AX.25 stations in the ordinary way. Using this interface, if the first character in a line terminated by <CR><LF> is <ESC>, the line is interpreted as a command, rather than data. If the first character of the command itself is a digit, the selected channel is changed accordingly and remains in effect. In addition, a rather ad-hoc set of editing rules for <CR> and <LF> had to be implemented for compatibility with local BB stations, so this interface is necessarily code sensitive.

In the case of IP datagrams, a full-bore IP-address/callsign dynamic mapping cache was implemented using two tables, the Channel Table and the Route Table. The Channel Table latches the current status (state and time-to-live (TTL) counter), last non-data message received and callsign. The callsign is used as the local-network address in the IP model. Following is an example (slightly edited) display captured during normal operation:

```
CID  CallsignStatusMessage
-----------------------------------------------------------------------------
0    W3HCF   0  05 fm W3IWI to KA3DBK via WB4JFI-5* ctl I17 pid F0
1    W4HCP    0  03 (1) DISCONNECTED fm W4HCP via WB4JFI-5
2             0  00
3    W3HCF   2  1200 @PF0
4*            0  01 CHANNEL NOT CONNECTED
```

The CID field indicates the channel number, with the asterisk indicating the currently selected one, while the Status field is coded as first the state and then the TTL. The intent is that, if the TTL counts down to zero the channel is closed. The reason for this will be described momentarily. The Message field begins with the <code> field (codes 1-5 only) of the last message from the TAPR followed by the <text> field as-is. as received. A code of zero indicates the last message was a command sent to the TAPR. In the above, channel 2 has never been used.

The Route Table is maintained as a cache using mechanisms similar to the Address Resolution Protocol (ARP) used with Ethernets. It consists of an associative array of two-field entries, one field containing the IP address and the other the associated digipeater route. The table is presently handcrafted; however, plans are to dynamically manage the entries as the result of monitored channel activity, perhaps augmented by ARP-stype broadcasts. Obviously, this is where the fun lies in the protocol-development area. Following is an example display of the current version:

```
IP addressRoute
------------------------------
[128.4.1.1]W3HCF WB4JFI-5
[128.4.1.2]W4HCP WB4JFI-5
[128.4.1.3]WD5DBC WB4JFI-5
[0.0.0.0]W3HCF WB4JFI-5
```

Note that the contents of the Route field can be used as-is as the argument in a TAPR connect command. In this particular case, the local radio channel is configured as subnet 128.4.1 of net 128.4, which is the swamp in which our own fuzzballs sleaze. Also note the default [0.0.0.0], which matches all other IP addresses, used for loopback and testing.

Connection Mode

The whole contraption can operate in either connection or connectionless mode. In connection mode it operates like this. Assume no channels are connected and a datagram originates at the fuzzball or arrives from another link destined for 128.4.1.2, previously determined to belong to W4HCP and to be reachable via the WB4JFI-5 digipeater. The Route Table is searched for 128.4.1.2 and yields the route string W4HCP WB4JFI-5. Next the Channel Table is searched for an active channel matching the destination callsign W4HCP in that string, keeping track of the highest-numbered inactive channel in the process and avoiding the monitor channel (0) and the currently selected ASCII channel (4 in the above display). If the match succeeds, the datagram is immediately sent on the associated channel; if not, a connection command specifying the route string is sent on the higest-numbered inactive channel and that channel is marked active. The datagram starting all this is sent following the command and is queued in the TAPR. Note that all datagrams are sent with protocol id CC (hex), which is assigned to the IP network layer in AX.25.

Datagrams arriving from the TAPR on an active channel are simply tossed into the normal fuzzball routing algortihm just as if they arrived from other links. A transmitted or received datagram causes the TTL field in the Channel Table entry to be reset to its maximum. The TTL fields of all active channels are decremented once in a while and, if one of them counts down to zero, a disconnect command is sent to the channel, allowing it to be used for possibly other destinations. Note that the AYT mechanism mentioned above is also operative and would normally catch a broken station before the TTL mechanism did. The TTL mechanism is intended primarily as a garbage collector when channels lie dormant for unreasonable periods.

Presumably as the result of monitoring activity, amendments are continuously being made to the Route Table. If a connection breaks with datagrams queued, the queue will be discarded and a new connection attempted according to the latest information in the Route Table. The lost datagrams will be recovered through the ordinary retransmission mechanisms of the client transport protocol. Obviously, the key to the success of this scenario lies in the ability to effectively collect the route information and maintain the Route Table. Further development is anticipated in this area.

Connectionless Mode

In connectionless mode all IP datagrams are sent and received on channel 0. This requires considerable heroics in separating the desired IP traffic from other traffic on the channel, not to mention monitoring headers and data. A datagram originating at the fuzzball or arriving from another link initiates the same operations as in the connection mode, as described above, except that channel 0 is used and it is never marked active. This implies some overhead, since a connection command is always sent preceeding every datagram. It is easy to avoid this by the simple expedient of saving the last argument and suppressing this if the argument is unchanged.

At present the only mechanism to capture IP datagrams from the radio channel is by turning on UI-frame monitoring (Ethernetters know this as promiscuous mode) and carefully filtering IP datagrams from the general slosh. The Thursday-night implementation (which should be obsolete by the time you read this) is to toss everything at the IP-header validation routine, which verifies correct IP header format and checksum and with high probability chucks everything but IP datagrams on the floor.

Initial Experiments

The driver is built so that ordinary monitoring headers and data are sent to the system log file, which allows channel activity to be observed in real time, as well as captured for later analysis. During initial testing it was obvious that a fair number of other protocol implementations active in the local community are defective, inefficient or worse. A report on that will be presented at another time. Specific IP tests were conducted in connection and connectionless mode via the WB4JFI-5 digipeater, which is on a high tower and has a very strong signal in the W3HCF local area. It can then be assumed that almost all packet losses are in the direction from W3HCF to WB4JFI-5 and those due to collisions aggravated by the hidden-terminal problem.

Initial experiments used virtual-terminal (TELNET), file-transfer (FTP) and raw-datagram (ICMP Echo) protocols. In general, the tests were successful and demonstrated that the Internet protocols worked handily via the lashup. However, several problems were immediately apparent, among them:

1. The speed of the fuzzball-TAPR connecting line is only 1200 bps, so that nontrivial queuing delays due monitoring and status information place exceptional strain on the TCP retransmission-delay estimation algorithm. The result is an unneccessarily high incidence of TCP retransmissions. Obviously, the cure for this is to hop up the speed, which will have been done before you read this.

2. In connection mode when times get tough the TAPR tries heroically to get the traffic through, which may involve several minutes of retransmissions, resets, etc., during which TCP is retransmitting and possibly abandoning the connection. Even if the traffic does get through, the result is gobs of useless duplicates and end-end ACKs which clutter up the channel. The cure for this is to reduce the max-retransmission limits and short-circuit the reset procedure.

3. I have observed that, on average, between one packet in four and one packet in ten is lost on the local radio channel. This is a high loss rate even for TCP, which is designed for a more lossy environment even than HDLC, let alone AX.25. Casual inspection of the monitoring log shows inspired braindamage on the part of some local AX.25 implementations using excessively large window sizes, which lead to excessive retransmissions at such high loss rates. This problem has nothing intrinsically to do with IP or TCP and can be cured by judicious intervention by the Protocol Police.

Conclusions and Plans

The "Nagle Algorithm" in conjunction with other refinements developed for the fuzzball TCP implementation was designed specifically to minimize spurious packet traffic and in principal could result in much improved performance. While the packetization and estimation algorithms designed for the present generation of TCP implementations have been designed for an extremely wide range in delays, speeds and loss rates, they have not been optimized for the extreme regime represented by a 1200-bps packet channel with loss rates in the neighborhood of one in four. The fuzzball TCP implementation, which has been designed and heavily instrumented for protocol development, is a useful tool to explore these issues.

>From a purely engineering perspective the use of ARQ techniques in such a regime is flawed at best and should in any case be augmented by some form of FEC coding. In addition, the use of compression techniques (e.g. the SLIP

protocol) should be explored as a mechanism to reduce packet-header overheads. Exploration of these issues may lead to revised packet formats and checksumming procedures. One of the first results expected to come out of this investigation is a requirement for separate checksums on the link-level header (AX.25 address, route and control bits) and data portions of the frame. The expectation is that residual "errors" alleged at the AX.25 level would be corrected by its client while nevertheless preserving the integrity of the routing and framing functions.

Probably the most effective realization of connectionless mode in the AX.25 paradigm would involve implementation of at least the IP-datagram support in the TAPR or equivalent. This is not really hard at all and amounts to adding a feature, keyed on the presence of CC in the protocol-id field together with a specific destination callsign, to indicate an IP message to the client host, which should be delivered with AX.25 header intact in order to support dynamic route cacheing as discussed previously. It should be possible to specify a list of such callsigns, which could then be used to implement a multicast/broadcast capability similar to Ethernets. This feature should be completely independent of the monitoring facility. Finally, it will be important in future to implement a filtering mechanism, based on the protocol-id field, which allows the casual user to avoid screen clutter due to unprintable code combinations while monitoring non-vanilla protocols.

Dave

C2973 CC8 Jack Brindle (WA4FIB,2963) 10/21/85 12:11 AM L:42
KEYS:/DATAGRAMS TCP IP/

Phil; After reading your lengthy memo, I have some comments. After noting the new report here, I think I will reserve many of my comments until afterr I can read it. However, I would like to ask a few questions of you.

In the TCP/IP environment, are you envisioning several levels of network, of the local, regional, national levels? Otherwise, Does each tnc require an International Electronic Callbook to be able to route packets?

I have some problems requiring each tnc to have layer 3 implimented. To put it another way, how are you going to explain to literally thousands of Kantronics owners that they can no longer participate in packet radio because their tnc does not have (or for that matter, has no chance of) the AX.25 network layer protocol???

After blasting you, I should offer some alternative. Well, it will be forthcoming, Real Soon Now... I would like to get a chance to further study your proposal before really commenting on it. I am curious, though, as to the exact nature of your current project and its relation to the memo you loaded on the system (apparently off ARPAnet). You previously mentioned your implimentation of IP. Are you also doing TCP as would be required in a network? (If not, then what good is IP?) And, does the code run atop your layer 2 code? I would be very interested in obtaining a copy of your C source (to run on a Mac, of course - I have so far resisted obtaining an 820 although I am beginning to see it coming...). Whether or not it will be the "chosen" protocol is rather immaterial now, since it is the only game. It does give us the opportunity to play with networking, however, to demonstrate to the "masses" that we are actually doing something.

On another subject. I am looking at doing a 68000 based NCC. The current design has 3 8530 SCCs (1 polled/interrupt, the other 2 DMA), a 6522 VIA (or 2) and 20 28 pin sockets for byte wide memory chips. Four of these chips would be located at low memory, and would be accessable at all times by the cpu, the remaining 16 would probably be ram accessable both by the cpu and dma controller. The dma controller has not been chosen at this time, but I don't want to use the Motorola dual channel chip. I am also looking at a time of day chip of some sort, probably serial. This looks to be a rather powerful system, and is not necessarily meant to compete with the TAPR NNC, but rather to give us a more powerful controller for higher speed use. If you have any comments or suggestions, pass them along. My feeling on this board is to allow the user to "mix and match" the peripherals for his needs.

More later, after I read the long message. I also need to figure out this system more, like just how to respond to someone who sends me a (an) interactive message. Maybe I need the eies manual after all!

73, Jack B.

C2973 CC9 Phil R. Karn (ka9q,2979) 10/21/85 6:41 PM L:69
KEYS:/NETWORK PLANNING/

Jack,

Dave Mills' work is complementary to my own. He is trying to gain some early experience with running the Internet protocols on top of AX.25, using off-the-shelf components at each layer (existing TNCs and Internet hosts). My job has been to design new implementations for AX.25/IP/TCP that are designed to fit together more cleanly on a small, standalone machine. I wrote the AX.25 code with the necessary software "hooks" to the bottom end of an IP layer, and if it can all fit on a single machine (and it now appears that it can) then the interfaces (especially the one AX.25 presents to IP) will be much cleaner than the ones Dave is struggling with.

The structure of the network (regional/local/etc) is a question of addressing and routing, not the protocols. In any event, my second conference paper (in the 4th Proceedings) gives my thoughts on this subject. Rather than repeat the whole paper here, I'll just summarize my main point.

For a network of reasonable size, we need a hierarchical addressing and routing scheme that also permits a somewhat arbitrary topology (e.g., ad-hoc satellite links). I am proposing an addressing scheme that is a hybrid between the complete topological flexibility of flat addressing (where every station has a complete list of every other station, so that topology changes don't require readdressing) and a hierarchical scheme where (most) addresses depend on the network topology in such a way that routing table space is conserved. I would welcome any comments on this paper.

The end-user nodes (the "hosts" that speak TCP) need not implement full-blown routing mechanisms. A common strategy on small machines (e.g., IBM PC's) that speak TCP is to send all traffic destined outside the local area to a "default gateway". The gateways are of course full-blown IP packet switches, and contain complete routing tables. It is possible for there to be several gateways out of one local network, so if the gateway is given traffic that is more optimally routed through another gateway, it may inform the user host of this with an ICMP Redirect message. In this way the user host may build up a small routing table of its own, but it need contain only those sites with which it actually communicates, and only if these sites are best reached through a gateway other than the default.

I understand your concern about thousands of existing Kantronics TNCs. I cringe every time yet another manufacturer comes out with a box just barely able to squeak by at 1200 baud with AX.25. Packet radio is a rapidly evolving field, and it is clearly shortsighted in the extreme to design units that barely operate satisfactorily with current standards, to say nothing about leaving room for future developments. In any event, it isn't totally clear yet that TCP *won't* fit on such a TNC, given the reasonably small size of my implementation and the exponentially increasing capacity of EPROMs. It is always possible to put TCP in another box which uses the TNC as just a link controller, in the same way that people plug smart terminals into simple modems. One of the things I hope will come out of Dave's work is a set of recommendations for level 2 TNC designers on the interface it should present to an external IP client (as opposed to a dumb terminal) over the asynchronous port. This should make things easier should the TNC manufacturers decide not to support the whole package in one box.

Our biggest mistake in amateur packet radio is that we have been building the networks from the bottom up, instead of the top down. For example, we have formally adopted only a link level protocol, but because every network must have a transport and network protocol we are in fact using AX.25 for those functions. Changing over to "the right way" can be pretty painful, especially when amateur packet radio has been growing the way it is (way too fast for its own good, I might add.) In retrospect, we should have first established a standard TRANSPORT layer protocol (i.e., TCP), and then worked downward as we felt the need for more sophisticated network layer services (e.g., automatic routing) and link layer services (e.g., hop-by-hop acknowledgements). There was in fact a proposal from KA6M to do just this way back in 1981, but unfortunately it was run over by the AX.25 steamroller.

Phil

C2973 CC10 Phil R. Karn (ka9q,2979) 10/22/85 12:11 AM L:43
KEYS:/YOU THINK I GET EXCITED?/

From brian@SDCSVAX.ARPA Thu Oct 17 14:48:16 1985
From: brian@sdcsvax.arpa (Brian Kantor)
To: karn@mouton.arpa
Subject: you'll enjoy this
Status: R

Wish he had a ham license...

```
        From fair@ucbarpa.Berkeley.EDU Thu Oct 17 10:37:14 1985
        Received: by sdcsvax.ARPA (5.28/4.41)
                id AA00619; Thu, 17 Oct 85 10:34:48 PDT hops=0
        Received: by UCB-VAX (5.28/5.13)
                id AA13164; Thu, 17 Oct 85 04:51:12 PDT
        Received: by ucbarpa (5.28/5.12)
                id AA05983; Thu, 17 Oct 85 04:51:05 PDT
        Date: Thu, 17 Oct 85 04:51:05 PDT
        From: fair@ucbarpa.Berkeley.EDU (Erik E. Fair)
        Message-Id: <8510171151.AA05983@ucbarpa>
        To: brian@sdcsvax.arpa
        Subject: Re: IP/TCP bumps and grinds
```

Actually, I do know that IP is usually encapsulated in something. Even in Rick Adam's SLIP...

As for ISO, fuck 'em. They haven't produced ANYTHING real. Read Padlipsky's book (or if you're really cheap, read RFC871-875, since they comprise the majority of his book, modulo some very interesting, nd irreverent comments).

The point is that X.25 is a crock of shit, and IP is the right way to go. It is trivial to provide an attached network processor, (like Excelan, CMC, and a bunch of other board manufacturers do for the IP/Ethernet combination) in most of the existing ones, the only thing that the host must do is data demultiplexing (i.e. deliver data to the right processes), because all the other details are handled by the board. Of course, the boards also have to allow raw access to the IP module, and to the network itself, if it is going to provide full functionality...

It would (as I said before) be a damn shame to see packet radio go the wrong way.

Erik E. Fair   ucbvax!fair   fair@ucbarpa.BERKELEY.EDU

C2973 CC11 Howard Goldstein (N2WX,2987) 10/22/85 8:43 AM L:4
KEYS:/ACIDIC/SURPRIZE/
A: 10

Phil - such vitriole from Mr. Fair! Boy won't he (and all the disbelievers) be shocked when TNC 2's are delivered with **AX.25 Network Layer** installed and working!! 73 Howie

C2973 CC12 Phil R. Karn (ka9q,2979) 10/22/85 1:27 PM L:3
KEYS:/SURPRIZE/

Howie, would you be so kind as to tell us what "AX.25 Network Layer" IS?

Phil

C2973 CC13 Howard Goldstein (N2WX,2987) 10/22/85 2:42 PM L:2
KEYS:/SURPRIZE/NOT THE PLUMBING KIND/
A: 12

Phil - AX.25 network layer is X.25 level 3 sans PVCs. 73 Howie

C2973 CC14 Thomas A. Moulton (W2VY,995) 10/22/85 3:54 PM L:24
KEYS:/NETWORK PLANNING/YOU CAN BURN ALL THE PROMS/
A: 9

Phil,

If your intents get out to the people who have made packet radio so popular (aka Kantronics, Heath, AEA, etc)
We will be FUCKED just like you said back a couple of comments, but instead of it just being the CCITT jerks, it will also be the TCP/IP jerks and the rest of packet radio, and then we might as well go back to RTTY

I think you are saying that we MUST REQUIRE each user to buy "my" box frankly I think our networks should support ALL the people who have these old boxes, hell I'd like to see people using the Vancouver and Ashby boards!

To turn around and tell everyone that they have to sell their WORTHLESS TNC 1's and buy the new TCP 1 will KILL packet radio, that would be a little like telling us we had to use V.22 instead of BELL 212

What we see as a reasonable approach is to ALLOW users to have level 3 PADs but also support level 2 users by letting them connect to the switch, and being prompted much like the PADs the PDNs use (Telenet/Uninet) given time the older boxes will fade away, (like AM operation has) but we shouldn't just turn our backs on them, I think that I would feel this way, even if I did like TCP/IP!!

73, Tom

C2973 CC15 Phil R. Karn (ka9q,2979) 10/22/85 7:15 PM L:8
KEYS:/AX.25 LEVEL 3/

Howie,
I wonder if you are aware that X.25 is an INTERFACE protocol, not a NETWORK protocol? It describes only the means by which a customer gets into the network from outside; it says nothing about how the network functions internally. I think you probably mean X.75 instead of X.25, but even there issues such as routing are left completely unspecified.

Phil

C2973 CC16 Phil R. Karn (ka9q,2979) 10/22/85 7:31 PM L:33
KEYS:/WE BURN PROMS ALL THE TIME/

Tom,
I never said that we MUST REQUIRE each user to buy "my box". What I said was that we must require each user to speak a common transport protocol. I don't care what hardware they run as long as it can speak the proper protocol. As I think I've shown with my TCP implementation, it is small enough to completely dispel the old objections about code size.

Perhaps I have been a little too strong in my condemnation of Protocol Conversion Gateways. It is true that one can lash them together if supporting unmodified TNCs is absolutely necessary. However, it is very difficult or impossible to obtain the full benefits of the Internet style protocols by doing so. You can get "remote login" capabilities pretty easily, just as I can by dialing into my UNIX host from home with a dumb terminal without having to speak IP across the dialup line. However, that's about all you can do with a "vanilla" phone line, and if you've got something more powerful at home than a dumb terminal, pretty soon you start wishing you could get direct access to the underlying network services.

If you want to build such a "gateway", be my guest; all I'm saying is such a device must not be thought of as a permanent part of the network but rather a temporary stopgap to ease the conversion to true end-to-end transport protocol implementations. I maintain that this is true REGARDLESS of the specific transport protocol that is selected, and I think that even Gordon agrees with me on this point!

My own priority, however, is to build a solid foundation for our network FIRST, before we worry about building conversion stopgaps. If we always placed backward compatibility ahead of everything else, packet radio would be using 5-level coding at 45.45 baud!

Phil

C2973 CC17 Thomas A. Moulton (W2VY,995) 10/22/85 8:55 PM L:20
KEYS:/I LIKE TO TALK TO DX STATIONS/THEREFORE DON'T LIKE US DOD TCP/IP/
A: 16

I fully agree that the transport can be anything, my previous comment was mostly fueled from you comment on cringing when another vendor decides to promote packet radio and put yet another box on the market.

you keep screeming that we should use this or that and saying that X.25 level 3 is this or that, THIS HAS NOTHING TO DO WITH END-TO-END, just the links to get there, I am working on things, we will include routing, using standard addressing which will be based on good old area codes which we all have maps for, just open your local telephone book.

you said you want to get a solid foundation before you worry about the rest
we agree, but why are you doing the transport layer now?
hadn't we better get the end-point to end-point network up first?
that's what x.25 level 3 gives you, then we will write session and then TP 4
on top of it all, I still want to talk to our european friends without having
to go via a gateway, because information gets lost in gateways
and we know how the PTT's feel about protocols
(we're lucky they agree to international standards)

i don't know why i keep beating the dead horse

C2973 CC18 Jack Brindle (WA4FIB,2963) 10/23/85 1:31 AM L:81
KEYS:/THIS RING HAS NO CORNERS.../

Tom; Phil is working on the Transport layer implimentation with good reason. Since we currently have a point-to-point system, we have no need for layers 3 & 4. The transport functions are essentially handled by the link layer. The checking provided for by L4 is being done by the AX.25 L2 implimentation. And, essentially the link layer is handling what the network layer will be doing in the future. Phil is right in that we have a sort of datagram system within layer 2 to provide the relay functions. Since no connections are made a L2 except for end-to-end, it falls short of true networking in either the VC or datagram sense. By the way, one thing to remember is than in X.25, there are only two addresses used at the link layer! Our transmission medium required a major departure from X.25 on this point.

When layer 3 is introduced, we will need a transport protocol to maintain data integrity. The transport layer is more important to the datagram approach because of its use of multiple routes for packet delivery, and thus its tendency to deliver packets out of sequence. Both the Virtual Call and Permanent Virtual Circuit approaches require a good Transport protocol to assure data integrity. The protocol need not be as rigid as that used for datagrams (it would be nice, though). One point that has escaped many people is that packets going through a network do not necessarily go in sequence, but may become seperated by other user's packets. This may happen for several reasons such as buffer allocation overflows or simply from higher priority traffic. This one point will kill Phil's method of breaking datagram packets into AX.25 sized packets without proper sequencing (again remember that the layer 2 sequences mean nothing at layer 3!, or between the next two nodes in the network).

The point is that we do need a layer 4 protocol, and should be working on an implimentation coincident with the layer 3 work. My initial file transfer work involved layer 7,5 and 4 protocols until I realized that I needed checkpointing in layer 5 (yes, there really is a good use for the session layer). My current approach, and the one I am proposing, uses a session layer protocol to provide checkpointed, and thus error-detected, transfers, not only for files but also for any other type of QSO. The MacPacket implimentation will allow at least 8 concurrent sessions! The specifics of the file transport are handled in the Application layer.

This will allow disimilar (sp?) machines to transfer files. They may not recognize the specific formats (A PC would throw up on MacBinary files), but at least they could send and receive them so that the files can be reconstructed on a similar machine in the same form as they were sent.

Note that I can get away with only layers 2, 5 and 7 because of the functions each handles. If I were to allow multiple connects with several stations, layer 4 would be required to seperate the incoming packets and provide the other services the transport layer performs (resequencing, error detection, class of service, etc.).

One question I have for you and the others working on the VC approach involves the handling of priorities. Howie and I have discussed this, and have noted that it is one of the shortcomings of the Tery Fox L3 proposal. Because of the nature of data that we will be transferring, we will need to impliment priority in both layers 4 (through service classes) and layer 3 (through proper queue management to layer 2). The place to put the priority info is in the LCI (I suspect that few packeteers understand the significance of LCIs. For example, a packet coming in from one node and the same packet outbound to the next probably will have completely different LCIs!). My recommendation would be to use the top two bits of the LCI to provide a 4 layer prioritization scheme, that is probably enough for our use. We will need to set up rules for service classes and thus priority so that file transfers receive a lower priority than keyboard QSOs, which are lower than Voice QSOs.

We have a major problem with current TNCs (solved in the WA8DED code) due to their user interface. The fact that the tnc does not report the delivery of a pack (known from the received ack) hurts the upper layers ability to perform error detection (I don't believe DED solves this problem either). We have proven that computers do better with numeric interfaces than human type verbose interfaces. This has hindered my work in developing MacPacket, and has caused many upper level designers to wish for something better from the next TNC software release. The approach the that Ron, WA8DED has taken is interesting in that is resembles the IBM approach of poll/response. That has proven to be quite reliable in practice, and seems to also work well in the TNC1 environment.

Anyway, It seems that we all have a lot of things in common. I believe that Phil has some very valid points, and although I don't believe that datagrams are the best approach (until Phil can prove it to me), I also see that VCs have major problems (also remember, I support the use of PVCs with VC used only to connect a new node to the network on a temporary basis - until the network fully integrates the new node). I am quite willing to work with all those implimenting network software in whatever way I can - even to code it on the Macintosh (please, a C source first). This should be interesting, I can indeed see that the folks have sort of "shut up" while they prepare to "put up". Whatever emerges should be quite interesting.

More later. 73, Jack B.

C2973 CC20 Lyle Johnson (WA7GXD,2973) 10/23/85   1:39 AM L:50

KEYS:/HARUMPH/

A: 10

Gentlemen! Really!

.

(here I digress)

.

There are 14 counties in Northern Ireland, largely Protestant. The rest of the island is primarily Catholic. Both groups are utterly convinced of the rightness of their cause. They have been sniping at each other for a long, long time.

.

And neither side is winning.

.

(or was it digression?)

.

Since I am neither a protocol definer nor a software writer of any particular skill, I am a mugwumpper. (As you may recall in American history in the 19th century, the mugwumps sat on the political fence with their mugs on one side and their wumps on the other.)

.

As a mugwumpper, I am happy to see the progress Phil is making on TCP. 5k! That will fit on any TNC TAPR has produced (read AEA and Heathkit).

.
I am also happy to see Howie's comments on AX25 Level 3 on TNC 2. As I understand things, TCP should run just fine on AX25 L3 (or AX75 or whatever the VC level three is supposed to be called). That is to say, I have been given the impression that levels three and four are supposed to have some sort of well-defined interface so various transport layers can run on various network layers...
.
I suppose there are two other non-technical points I would like to state.
.
The first is that we must be careful to not take ourselves too seriously. Of course, the work that is going on now will have (and is having) dramatic impact on the Amateur landscape. But it IS amateur. The more closely we emulate the commercial approaches (which involved engineering compromises thought best for the COMMERCIAL systems), the more closely we will suffer their same weaknesses. Not too good for disasters, etc., where amateurs get the good will they need to not lose frequencies in Geneva...
.
The second is that amateur radio is thinly disguised anarchy. And that is a healthy state.
.
Packet is technically driven. We don't need emotional IRA-style approaches to "solving" our problems. Four-letter epithets merely indicate a lack of ability to reason and express oneself, a surrender to emotional reaction rather than rational thought.
.
Off my soapbox now.
.
Lyle

C2973 CC21 Phil R. Karn (ka9q,2979) 10/23/85 1:47 AM L:19
KEYS:/ITS MORE LIKE ISRAEL AND LEBANON/68K CONTROLLERS/

Ah hem.

Lyle, as a member of the Digital Committee, have you received a copy of the letter I sent to Paul Rinaldo on protocols? If you haven't, I can summarize the whole thing in one sentence: protocols are important, because they transcend all of the various hardware and software implementations that speak them. That's why its important that we hash out these issues NOW, before inadequately thought-out standards (I was tempted to insert the word "more" in front of "inadequately") are promulgated to thousands of users.

Regarding a 68K NNC: if I have a 68K with a reasonable amount of memory and DMA controllers for the channels, there is NO NEED for outboard processors to do the level 2 functions. They can be accomodated very easily in the 68K itself, and the interfaces are a lot easier to fit when it's all in software. Please don't over-design the hardware; it only makes things harder, not easier, for the software implementers!

Phil

C2974 CC100 Lyle Johnson (WA7GXD,2973) 10/23/85 1:53 AM L:26
KEYS:/NNC UPDATE/BETTER UPDATE TO FOLLOW/THE CABBAGE TRUCK COMES REAL SOOON NOW/

NNC update (sorry it is hasty and brief, will be mmore "formal" ina day or so
.
There was an error in the asrtwork, so the digiatl board and the floppy controller are now one week down (well, 1 and 1/2) and two weeks to go to prototypes. We will get three sets if we are lucky.
.
There was mass confusion on the modem bnoards, so they are just now being cadded. Not in time for southnet2 I'm afraid, on the modems.
.
Just to give you an idea of the magnitude of this project, the CAD costs are right at $9,000 and the 2or three digital prototype boards will cost arount $1,300. Not small potatoes!
.
I will try to post a memory map here in the next couple days so those of you who are interested in proting software to the board can do so.

.
I hope to verify the design and get the artwork modified, wetc, so we can turno out a reasonable number for developers before the end of the year. Ambitious, and may not make it, but that is the target.
.
Still no estimates of costs, but I will post these as soon as I can get a handle onm them. No, we aren't even thinking of recovering the development costs on this one, the 10 grand is support from those of you who bought TNc 2.
.
Lyle

C2987 CC219 Howard Goldstein (N2WX,2987) 10/21/85 11:42 AM L:4
KEYS:/ALMOST FORGOT THIS ONE/K IS FOR KONVERSE/
A: 208

Almost forgot to mention

K is for Konverse -- In Command mode (cmd:), a lone letter "K" is synonymous with CONVerse command.

C2987 CC220 Tom Clark (W3IWI,2976) 10/21/85 11:20 PM L:20
KEYS:/BLACK HOLE MODE/I MAY HAVE TRAPPED THE BUGGER AT LAST/

Howie: I think I caught TNC2 (1.1.1h) in the black hole mode. This time it gave a FRMR, here is the dump (typed manually, since I didn't have a rcv capture file open and was only able to dump it to the printer):

```
FRMR sent:AEDE08
FRMR rcvd:
000: AE6692AE 924062AE 6692AE92 4062AE66 W3IWI 1W3IWI 1W3  .f...@b.f...@b.f
010: 92AE9240 E187AEDE 08                IWI pCWo.         ...@.....
*** CONNECTED TO W3IWI-1 VIA W3IWI
```

In this case, I was in TRANS and had pretty well filled^@the input buffer. Was running MAX 7 and FRACK 1 and everything finally went mute. Even though I was getting no echo, I did a ^C^C^C, set FRACK 5 and then got the FRMR. Again, the only way I can force it to happen, and then only some of the time, is when FRACK is short.

Hope this helps -- Tom

C2987 CC221 Howard Goldstein (N2WX,2987) 10/22/85   6:00 PM L:26
KEYS:/TOM/LYLE/SKIP/BLACK HOLE/CTRL Y/CONV DATA LOSS/

TOM - Ran into something similar to the black hole with the 10.0 mailbox terminal emulator and XFLOW OFF.

With 7 or 8 digipeaters, max 7 and frack 1 the buffer fills rapidly (in transparant with pactime e 1) after a few lines everything freezes up for what feels like forever. It does recover (not like the black hole problem) later when 8 digis end their convulsions and counters start getting updated.

Using the rli B command to send a file while self connected causes a lockup in the both-busy state eventually.

Are you using a terminal program which suppresses characters from the TNC while the flow control is active (like the mailbox)?

LYLE - ctrl-Y tends to work very slowly, especially at high async baud rates. will try to fix it

If possible check the new bank of memory, and/or the checksum for the 1.1.1 version giving you problems ($2f and $7f are both valid early and late checksums).

Using pins 3 and 7 on the rs232 was able to monitor the channel and accept connections okay. Note that the 1.1.1 releases enable hardware rflow always; in default XFLOW ON mode earlier releases used to ignore the state of CTS* on the SIO. The symptoms you mention sound exactly like how some terminal programs act when RS232 DCD isn't true all the time... Haven't seen the inverted CON led problem on the beta board, has anyone else?

SKIP - Still looking, also interested in CPAC/CR/SENDPAC settings during the data loss problem.

73 Howie

C2987 CC222 Lyle Johnson (WA7GXD,2973) 10/23/85 1:46 AM L:20

Howie,

.

I blew it. The problem is the STA LED. Eric connects, has outstanding packets (STA on), then enters cmd: mode and does a pair of rapid Disc. The STA LED stays on. Now, if he connects to another station, the STA remains out of sync. Sorry about the earlier misinformation.

.

Eric is using release 1.1.1 Checksum $7F, 16k bbRAM. If the upper RAM isn't good (lower? the one in the middle socket) it doesn't work at all, so the RAM is good I think. It is a RAM I used as the only 8k RAM in a TNC 2 I was using, so my confidence in it is pretty high. Incidentally, we have about 550 1.1.1 $7F EPROMs programmed now...

.

He is also using a CoCo with 3-wire interface. I will verify that it isn't a four-wire one with some silly flow parameter, but it worked fine with 1.1.0 release version software.

.

Cheers!

.

Lyle

M 6387 J. Gordon Beattie Jr. (N2DSY,2990) 10/23/85 4:01 AM L:11
KEYS:/SOFTNET/56KB MODEM/9600 BAUD MODEM LINK REQUIREMENTS/
TO: (Group 95)

---

Who made reference to Softnet ? Is there any further information on it ?

---

Anyone have a 56kb modem planned or working ?

---

What is a typical 220 MHz circuit configuration for the K9NG 9600 baud modems ? What radios, power levels, preamps, coaxial cables and antennas are being used to cover what kind of path ? What kind of performance is being obtained ? Is it a two point circuit or a multipoint circuit ?
Thanks, Gordon, N2DSY

M 6720 Lyle Johnson (WA7GXD,2973) 10/24/85 12:49 AM L:8
KEYS:/SOFTNET/
A: 6387 TO: (Group 95)

Gordon,

I mentioned it. I will try to get an address for you. They have a multiple 6809-based distributed forth (as in the language) system linked on 402 MHz (non-amateur, and maybe non-existent, but very experimental) with 100 kpbs modems (the modem/radio is the most iffy part).

Lyle

M 6754 Tom Clark (W3IWI,2976) 10/24/85 2:18 AM L:16
KEYS:/PACKET RADIO IN ETHIOPIA/QSP DE WA9FMQ & VITA/
TO: (Group 95)

The following is re-transmitted from AMSAT's TeleMail network:

Posted: Wed Oct 23, 1985 11:56 AM GMT
From: VITA
To: AMSAT, PACSAT
Subj: PACKET IN ETHIOPIA
LATE WORD FROM OUR VITA VOLUNTEER IN ADDIS ABABA IS THAT GOVERNMENT APPROVAL FOR A DEMO PACKET PROJECT/FEASIBILITY STUDY HAS BEEN APPROVED!
AS SOON AS MORE INFO IS KNOWN, I WILL BE CONTACTING THOSE WHO EARLIER THIS YEAR EXPRESSED INTEREST IN GOING OUT THERE FOR A FEW WEEKS TO GET THIS MOVING.

IF ANYBODY ELSE IS INTERESTED IN DOING A VOLUNTEER (BUT EXPENSES PAID) STINT IN ETHIOPIA ON BEHALF OF PACKET AND EMERGENCY COMMUNICATIONS, LET ME KNOW ASAP.
GARY/VITA (WA9FMQ)

C2973 CC22 Phil R. Karn (ka9q,2979) 10/23/85 3:45 AM L:105
KEYS:/TCP ISN'T CLASSIFIED YA KNOW/NETWORK DESIGN/

Tom: there's no law that says non-US hams can't speak TCP. The ARPA Internet extends to quite a few countries in Europe and to Japan and Korea. We speak TCP/IP regularly across a Telenet/Israeli PTT X.75 connection to Hebrew University Jerusalem (where one of our sabbatical visitors is from), and no member of the International Federation of Protocol Police has yet objected.

It's interesting that you propose addresses based on area codes. You say you want DX contacts, but our telephone numbering plan is limited to North America. I hope you've made provisions for other countries...

Jack: You are mostly correct in your comments regarding layer 2 ("internal") fragmentation of IP datagrams, As I mentioned earlier, AX.25 does not now have the ability to bundle several I-frames together as a single logical "packet", so something sitting above AX.25 level 2 would be needed to perform this function. One possibility is a simple "protocol" that uses reserved characters (with byte stuffing for transparency) to delimit the boundaries of datagrams that extend over several I-frames. Something called "SLIP" (Serial Line Interface Protocol) that does exactly this is already in fairly widespread use. We use it to send IP datagrams over ordinary asynchronous lines, since they also lack packet framing.

If a Level 3 PID were reserved for SLIP, and SLIP is presented with only those incoming I-frames having this PID, then things should work well. Since there's only one IP module atop SLIP performing routing and switching functions, there's no problem with pieces of different datagrams being intermingled. Other (non-IP) traffic having different PIDs can coexist on the same AX.25 level 2 link without interference, since it is the job of level 2 to route each incoming I-frame to the correct level 3 "module" based on its PID.

I'm not sure all this is really necessary, since we can either use IP level ("external") fragmentation, or better yet, allow stations to agree on larger level 2 packet size limits. IP is used to operating on top of many link layers often varying packet size limits. The usual procedure is for TCP to generate datagrams no larger than the maximum packet size of its local interface. If they later have to pass through an interface with a smaller limit, fragmentation occurs at that point and eventually the pieces are reassembled at the final destination; otherwise they arrive in one piece.

I'm curious to see exactly what functions you consider the province of the session level (layer 5). TCP guarantees end-to-end data integrity and provides application multiplexing and demultiplexing, so I don't know what else is needed. Application-specific higher layer protocols (e.g., FTP, the file transfer protocol, and SMTP, the Simple Mail Transfer Protocol) all have "well known" TCP port numbers, so TCP knows how to route incoming connect requests to the appropriate application servers.

Regarding priorities: this is something that IP and TCP do very well. The IP header contains a precedence and class-of-service field so that the network (which deals strictly in IP datagrams) can handle differing priorities on a packet-by-packet basis. The class-of-service field allows the user to control tradeoffs that might be available in the lower-level network used to pass the datagram. There are three bits: low delay, high reliability and high thruput. Setting one or two of these bits indicates that the corresponding service characteristic(s) is/are considered more important than the others. For example, setting the reliability bit might translate into the use of hop-by-hop acknowledgements on a packet radio net, or setting the throughput bit might cause a gateway to route over a high bandwidth (and high delay) satellite channel instead of a narrow but low delay terrestrial link, and so on. Precedence is also available at the TCP level, but this controls access to end-host resources instead of network resources: for example, during an emergency a certain electronic mail system might reject all connect requests below a certain priority level.

You comment about the lack of delivery confirmation facilities in existing TNCs. Here's where a datagram-oriented transport layer (e.g., TCP) comes in handy. Since TCP itself takes final responsibility for seeing that user data reaches its destination through the use of end-to-end sequencing and acknowledgements, it really has very little use for delivery confirmation from the lower layers. It's *nice*, of course, if the majority of datagrams sent by TCP do in fact reach their destinations, because that minimizes the number of TCP timeouts and retransmissions and hence improves performance, but it seems rather pointless to get a positive indication of delivery from the network for each and every packet when an end-to-end TCP acknowledgment will be forthcoming anyway. It also seems rather pointless for the network to go to heroic measures to absolutely guarantee delivery of every single packet come hell or high water (which it really can't do anyway), since it costs very little for TCP to resend it. The network makes a reasonable effort, and TCP mops up whatever falls through the cracks. As long as the number of datagrams that do get lost is reasonably small, performance is scarcely affected.

An analogy exists in the USPS. Very little mail is sent certified, and this is easy to understand when you consider that the surcharge for certified mail (and the extra complexity of the "interface", i.e., having to go to the PO personally) is greater than the cost of the stamp your correspondent needs to send back an ordinary letter saying he got your mail (and perhaps "piggyback" the acknowledgment to some information of his own, just like TCP.) Certified (or insured) mail is usually used only for items having a high intrinsic value (so "retransmission" is unusually expensive) or mail that is being sent to "uncooperative" receivers (e.g., jury summonses, although around here I get mine from the Sheriff's Deputies!).

The only time TCP can make use of advice from the network is when a datagram CAN'T be delivered for some reason. This is the function performed by ICMP, the Internet Control Message Protocol, and it typically causes TCP to abort a connection attempt (if it receives a "destination unreachable" message) or cut back on the rate it generates traffic (if it gets a "source quench" message indicating network congestion). It makes more sense to me to generate these messages only when something unusual occurs, instead of loading down the network with countless but redundant "all is well" notices.

Phil

C2973 CC23 J. Gordon Beattie Jr. (N2DSY,2990) 10/23/85 4:17 AM L:79
KEYS:/ON AND ON AND...WE HAVE PASSED THE PRODUCTIVE PHASE...SIGH/

In one shot here are my comments on the contributions of the last few weeks.

Jack:

I don't believe that the network will be flooded with a mass of level 3/4/5 connection setup packets.

Level 3/4/5 connections should be opened in one packet unless the upper classes of X.224 are needed in which case they must be done sequentially. This is not bad because the upper classes are used when more than one activity is being conducted between two points. An example of this is bidirectional file transfers and/or conversation.

I am also concerned about the hassle of specific-station PVC tables. This approach is not dissimilar to the hassle we have now with PBBSs needing message routing tables to every other system. An implicit routing scheme based on CCITT X.121, and in the case of the North American region, Area Codes will go a long way to simplify table maintenance.

The "Control" node in the mode you suggested sounds like a further burden. I'm not saying that having some node providing some centralized services is bad, but I don't want to depend on it for actual operation. I am afraid it either: 1. won't happen, 2. won't be there or 3. will be out of date. The security and assistance functions are welcome in such a box, but I don't want to get to a stage where we MUST HAVE ONE UP in order to operate. The download function is VERY DANGEROUS ! I have seen two nodes which had bad code correct one another and then proceed to "correct" switches with healthy code and tables. My suggestion would be to house the software and tables on a system and have the remote switch operator select the code he wants loaded into his machine.

May I suggest that you consider these issues when you think of your PVC based design.

Jack:

Transport protocol support belongs at the endpoints of a circuit. Dumb terminals could benefit from having this protocol support in the TNCs, but dumb terminals usually don't need heavy-duty error checking. Put the effort into a few of the common computers.

Phil and Jack:

The AX.25 Level 2 protocol is not in its point to point form a "datagramme protocol". Yes it has source and destination "addresses" in the header, but they are NOT there for routing purposes. They are there to uniquely identify a link in a common RF domain. In the context of the digipeater environment, the case is different, but again I don't approve (philosophically) of digipeaters either !

As one who has served the amateur community by helping define the protocol, I can say that in its preferable form (NO DIGIPEATERS !):

AX.25 IS NOT A DATAGRAMME PROTOCOL !

Phil:

Regarding your efficiency comparison: read my paper in the proceedings of the last networking conference. My position, and the facts hold up regardless of AMATEUR NETWORK PACKET SIZE. How big do you want Phil ? -- Save your breath: it doesn't matter !

Let us all understand one basic fact: the AX.25 protocol suite as we designed it (over three years ago) and as it is currently being implemented consists of a link protocol (AX.25 Level 2, version 2) AND a network protocol (CCITT X.25 Level 3, 1984).

Bravo to Howie in C2973 CC11 !

Brevity is not one of my stronger characteristics, but it seems that I've come farther than some...G'day !

C2973 CC24 J. Gordon Beattie Jr. (N2DSY,2990) 10/23/85 4:25 AM L:8
KEYS:/THE DOD IS LOOKING FARTHER AHEAD THAN SOME FOLKS CAN HANDLE/KEEPER OF THE DEAD SCROLLS/

4Phil,

I guess you need weekly reminders: The DoD is going to replace TCP with CCITT X.224. Let's all remember this and take turns reminding Phil.

There has been much discussion over painful conversions to "superior" protocols such as the DoD suite. Isn't it odd that the conversion from TCP to CCITT X.224 is NEVER mentioned in such discussions ?

C2973 CC25 Thomas A. Moulton (W2VY,995) 10/23/85 9:42 AM L:10
KEYS:/LEVEL 3 IS VERY MUCH NEEDED/
A: 18

Jack,
We have a best a shakey point to point protocol, DIGIPEATERS are a KLUDGE and should not be used much in the future, by running X.25 Level 3 we will get relaible point to point links, spanning as many stations as is needed,

not being limited to 2 + 8 digipeaters, the throughput will be higher bacause there will be hop by hop acks (between each switch) so no more long delays due to your frame being trashed on the first hop of 8 (or even 2)
If you get carried away you can call almost any protocol a datagramme, incorrectly
Yes there are two addresses, they specify the direction, in our case we must specify the call signs, A->B and B->A

C2973 CC26 Thomas A. Moulton (W2VY,995) 10/23/85 10:09 AM L:9
A: 22

What I meant by using area codes was to indirectly reference a paper from the last conference, 4 digits for country code and then the remaining digits are for national definition, in North America; area code (3 digits) and number 7 will work just fine, other countries use city codes and a local code before the number which will work also, after all we can call them on the phone too.

I do remember there were some countires that would only allow HAMS, (aka Amateur Radio Operators via RF) using internationally recognized protocols.

C2973 CC27 Howard Goldstein (N2WX,2987) 10/23/85 1:41 PM L:17
KEYS:/MULTIPLEXING/X.25/X.75/COMMERCIAL VC/
A: 15

JACK - X.25 network layer is especially nice at the user's TNC/PAD since it allows consistent way to multiplex different virtual circuits across a single link regardless of their destination. X.25 13 in particular is nice for this since it should be easy to implement on existing TNCs. Between two DTEs it also provides multiplexing (at least the way I have it.. what you're called as far as DTE/DCE is determined at call setup time)

PHIL -- As you saw in the TNC 2 beta test conference I don't have X.75 specs, so I'm interested in the differences between an X.25 interface implemenation and an X.75 network implementation. For what its worth - provided there's a routing mechanism - X.25 running between two DTE's seperated by n DCE's works quite well.

No one has yet discussed why the commercial networks use VCs. Does it have something to do with the cost of providing useful and reliable service?

73 Howie

C2973 CC28 Phil R. Karn (ka9q,2979) 10/23/85 3:36 PM L:53
(ORIG.) 10/23/85 3:19 PM L:53
KEYS:/I HAVE PLENTY OF PATIENCE/

Gordon, I don't want to get too bogged down in semantics, but I suggest you read chapter 5 of Tanenbaum's book (Computer Networks). A datagram protocol is one that has both the source and destination addresses on each packet; that's it. The link header of Ethernet is about as similar as anything you'll find to the lower part of AX.25, and everyone certainly considers Ethernet to be a datagram network. You are just bothered by the irony of having participated in the design of a datagram protocol, but a rose by any other name...

My point about header efficiency is to show that there are two variables in the equation (header size AND data size) and that the effect of a larger header can be made negligibly small (but not zero) by increasing the data packet size.

I thought I posted the DoD position paper on TP-4 vs TCP a few months ago. This was in response to the NRC report. Their position was that it is currently infeasable to specify TP-4 as a co-standard with TCP, or even to set a specific date at which this could happen. The reason? The standards are full of unresolved holes and missing functions, and commercial implementations are nearly non-existent. When (or if) they are plugged and TP-4 becomes available in something other than paper, DoD will reconsider using it and so should we. In the meantime, I think TCP has quite a few years of life left in it.

Gee, it seems to me that if we specify TCP as the standard transport protocol for amateur packet radio, then it will become "internationally recognized" in the same way that AX.25 did (the name is really a misnomer, because it isn't a compatible form of X.25 in the sense that BX.25 is). Or are there countries that don't permit the use of AX.25 TNCs because CCITT didn't publish the exact spec? I think a much bigger problem with other countries is the use of automatic repeating and linking by ordinary hams; certainly they could care less about the specific protocol.

Howie: commercial public data networks PROVIDE a virtual circuit SERVICE; they do not necessary use them internally. Telenet and Tymnet do, but other networks (e.g., Canada's Datapac) use their own datagram protocols inside the network. Why do the PDNs provide only a VC service? A couple of reasons. First, the protocol was decreed by the CCITT, an "old-guard" organization of telephony people who have had the "circuit switching" notion ingrained into them from day one. Second, 99+% of X.25 users do nothing with it but connect their dumb terminals through PADs and access their big IBM mainframes. In such a limited application ("remote access" as compared with the more general "resource sharing", which is characterized by more computer-to-computer kinds of applications) the X.25 straitjacket isn't much of a hindrance and the lower per-packet overhead is an advantage because most people program their PADs to send single-character packets. If you try to use X.25 for host-to-host (i.e., computer-to-computer) applications, as we do, you start running into X.25's limitations very quickly. You are forced to use more sophisticated higher layer protocols on top of X.25, making many of the "features" of the latter so much excess baggage.

Phil

C2987 CC223 Howard Goldstein (N2WX,2987) 10/23/85 8:46 PM L:12
KEYS:/FLOW/INDUCED FAILURES/
A: 222

Lyle - The earlier lettered releases assumed software only flow control until told otherwise. Wasn't TNC 1 compatable...$7F does use hardware flow all the time. It shouldnt affect echoed stuff though and any mode that disables the TNC echoing is bogus

I tried screwing up the 2 ram chips a few different ways, the least damaging of which (no RAM at 0xC000-0xDFFF) left the TNC flashing its lights and keying the PTT smartly - to the beat of distant drums, no doubt - and it echoed, but otherwise showed no signs of life in terms of prompts or anything else. The other induced "failures" left the board completely dumb (1.1.1 $7F 16K)

Feel rather strange with 550 blown chips and 2-3 ?bugs...

73, Howie "There's another bag of coffee in the cabinet over the sink" G.

C2973 CC29 Phil R. Karn (ka9q,2979) 10/24/85 3:14 PM L:19
KEYS:/WHAT THIS DISCUSSION NEEDS IS A GOOD DOSE OF COMIC RELIEF/

I forwarded an archive of the Protowars for the past week or two to Brian Kantor, WB6YCT out at UCSD. I thought you might be amused by his reply:

From brian@SDCSVAX.ARPA Thu Oct 24 14:41:16 1985
Date: Thu, 24 Oct 85 11:20:44 PDT
From: brian@SDCSVAX.ARPA (Brian Kantor)
To: karn@mouton.arpa
Subject: Re: EIES packet radio protocol discussions

You are hereby morally encouraged. Also immorally. Remember, we're right and they're wrong, and they're also too uninformed to know that they're wrong. Goodness and Right shall triumph over Evil.

Feel better? When I've got .3 second to myself I'll try to marshall some arguments, but you seem to be carrying the ball quite well.

Illegitimus non Carborundum est.
- brian

To All;

After watching DR Net for several weeks, I jumped in and restarted the protocol arguments specifically to bring myself up on the current thinking in the various "camps" around. I have noted several things. First, there are two groups, one centerred around datagrams chiefly represented here by Phil. The other supports Virtual Call type virtual circuits (while disavowing PVCs), and contains Tom, Howie and Gordon. Both sides have legitimate arguments, and they certainly like to argue. I wonder whether the two groups fully understand the stands the other group takes, and more importantly why. It is interesting, though. I just want to make sure that we don't go back to arguing instead of continuing on our projects.

Having said that, I will now dive back into the discussion. First off on the 68000 board issue. I am not averse to using an outboard processor to perform individual layer functions. This might be necessary if we want to attain high throughput rates (say T1). However, there is a major problem with current TNC implimentations. The WA8DED code is the only code available at present that has a decent computer interface, and runs in an "off the shelf" TNC. We continue to develop TNCs with dumb terminal type interfaces, yet the vast majority of packeteers have their TNCs attached to computers. It would be far better to allow the computer to perform the higher level functions and have the TNC perfrom the layer 2 functions it is intended for. If you wish to investigate this, try interfacing a Kantronics or TNC2 to a computer. You will quickly find out why I am making major changes in my MacPacket products (no announcement yet on that).

To be specific on what I would like to see, first I would like an interface that allows me to poll for data and status. Take a look at the way floppy disk controllers work. We should follow this example (This is one of the reasons that the Mac is so nice to program, its drivers are set up logically in this format!). Instead of throwing out data uncontrollably, the TNC should only speak when it is asked to. This also allows for a better seperation of commands and data. I should also be allowed a free dialog of the TNC's status - like when it sends a packet (or even better, when it receives the ack for that packet). This allows me to better perform the upper layer functions. For example, it is nearly impossible to add priority to a current TNC. (Yes, I know this is a layer 3 function). Since current TNCs buffer data from the host, they may have many packets stacked on their outbound queue. If the system wishes to send a higher priority packet, it can only tack it on to the end of the list. One way out of this would be to only send one packet at a time to the TNC. But I would need layer 2 delivery confirmation to perform this function. Thus we are back at square 1. Well, how important is priority. Suppose two stations are in a QSO. They have my session layer code that allows multiple sessions. One operator requests a file transfer. The other agrees, a new session is set up and the file transfer proceeds. With no prioritization, the data file fills up the outbound data pipe. The two operators continue their QSO, but find the going rough as their exchanges must wait in line behind the file transfer. A better situation occurs if the QSO is given higher priority than the file xfer so that packets from the keyboard are enqueued at the front of the line so that the QSO proceeds almost as if the file xfer were not happening.

Another reason for needing delivery confirmation at each layer is data integrity itself. Each layer must hold the outbound data (or at least a pointer to it) until it receives an ack from the peer layer. For example, suppose I send a packet. It comes in at layer 7, and proceeds downwards. Layer 4 adds its info, then passes it on to layer 3, etc. If L4 were to delete its copy of the data after it was sent to L3, then what would happen if the delivery were not made? Thus it needs to keep at least a pointer to the packet text, and the packet must remain intact, until a peer confirmation is received. This peer confirmation, of course, comes from the layer beneath the current layer. The lower layer also must report delivery to show the upper layer that it is functioning properly. Note that in our point to point system, this acknowledgement is equivalent to an end-to-end ack, and would release the packet as far as L2 is concerned, and would allow the session layer to also release the packet (Confirmed packets would also require a peer layer ack). The confirmation in a network system would simply show delivery to the next node in the network. It is important to realize, though, that these confirmations occur between each successive layer.

This also points out why it would probably be better to build the protocol for each layer into the node controller. The node must operate in a homogeneous fashion or we will have problems from a loss in data integrity. If we can get a system using NNCs as front ends for a larger processor, that is fine. In fact, that is the approach used by the larger Noed controllers (reference Paradyne's Pixnet XL line with processor boards using Z8000s for each layer). My 68000 NNC design does not preclude this approach. In fact, its memory design could even handle twin 68000 coprocessors interactively sharing the bus (what a programming nightmare).

Next subject, long packet fragmentation. Why not just shorten the packet length to a suitable size. We have a unique transfer media. Ethernet systems using Coax cable do not have the noise, QSB, and QRM problems we have with radio. Thus they can get away with long data packets. We have BER problems as it is. Try a test sometimes over a marginal, but useable, link. Shorter packets get through far better than long ones. The reason, of course, is that short packets have less of a chance of getting blasted than long ones. This is one reason that the 8 call digi field is idiotic. That excess baggage causes extra problems in getting good packets through. The answer is not to lengthen packets, but to either condense them (this is a good use for layer 6 - to compress data strings by, for example, changing multiple occurances of characters (a row of blanks) into a count & the char). We can actually increase our throughput by shortening the packets! AX.25 has set a max of 256 octets. This seems reasonable to me. In fact, the MacPacket file transfer protocol blocks data at 128 bytes. Add to this overhead from the various layers, and we are up towards 200 bytes per packet. Of course, the packet shrinks as we remove digis! This information would have a far better chance of getting through a marginal link than a packet with 1024 byte data blocks. Nuff said.

We all agree that digipeaters are an abheration (sp?) on the packet scene. Well, are they? They certainly have made our current operations useful. The big problem is that they use the stupid digipeater fields instead of real network routing. Well, folks, that is why we are here arguing. I suspect, though, that even after we get our networks up we will still have a use for the digis. They are very useful to link local stations that are just out of range of each other. For example, I can connect to packeteers all over the Florida west central coast using an IC2AT running 150 milliwatts, and a 1/4 wave whip antenna. Why? Because KC2FF-7 is only 1/4 mile away! Without it I would not be able to operate on packet! It wouldn't make sense to use the resource of the big local node just to talk locally. Even better, we may need the things to allow stations that are outside the rangge of a network node to access the network. I suspect this will be their real use. It will be interesting to watch as these things develop.

One thing that we should all remember is that we must make things easy for present and new users to play with the system. We have a lot of new packeteers who would not be on if not for the Kantronics TNC. These folks have decided to buy a new gadget and find out what the noise is all about. If we build a network that is difficult to use, or requires major modifications to their station equipment (The very simple Kantronics units may not be able to handle much more), then a separate, parallel system will sprout. This, of course, would undermine all our efforts. We must leave a place in our systems for these people. Their TNCs may eventually go away, but it will take quite a while. The vendors are not designing state of the art stuff, but they are making money. One example, the new Kantronics Version 1.6 ROM does not speak AX25L2V2. Instead it just rehashes the old code. They did add a "check" function which, in the absense of activity on the connection, will send SABMs to the other station. Prepare to see lots of "Connected to..." or "Link Reset..." messages caused by people using these units. But, to allow the use of these things, we must design the network interface for them. If I have the full network code in my machine I should be able to use it. Otherwise, I should be presented with information to allow me to use the system.

I foresee an interface that would present the Kantronics user with a menu for the routing function, and perform layer 4 functions for him. You might sya, "What a kludge", and you are right, but it would work.

Lastly, a comment on TCP/IP/Ethernet. The Ethernet protocols do not follow the OSI model. I understand that there has been an attempt to bring TCP/IP under the OSI model, but I do see that it did not completely happen. For example, XNS "sockets" (which I presume also exist in TCP) are actually OSI layer 5 functions. Thus, to answer your question Phil, You may not need my session layer. But you had better have its functions! The TCP/IP functions overlap the OSI model in many areas, creating confusion for thos not understanding both systems. In Xerox XNS/Ethernet, layer 0 handles OSI layers 1,2, and part of 3. This layer is known as the Ethernet layer. XNS then handles internet routing, the transport and higher OSI protocols. Of course, it still maintains an application type layer where the user system interfaces. I assume that TCP/IP is not much different due to their similar roots.

For those wondering about my session protocol, it allows multiple sessions within a computer. This would allow a multi-user computer to carry on a seperate session for each user on the system talking over the network, or allow seperation for many stations logged into a multi-user bbs. It also allows for multiple file transfers and qsos on MacPacket system,s. It works to seperate different dtat streams, so that whether the lower layers can handle multiple connects or not, the session layer allows that. When connected to one station, the user can both QSO and send file(s) simultaneously. When connected to sebveral stations, the user can talk to each, and transfer files simultaneously. It may be slow at 1200 baud, but it works. The streams are seperated into sessions, which are identified by a unique session id. The session layer relies on the lower layers for packet data integrity, but adds checkpointing in the case of file transfers. This is to ensure that file blocks are properly sequenced (this would seem to be a layer 4 domain, but layer 5 is uniquely set to perform this function for each individual data stream. Layer 4 sequences for all inbound packets, while layer 5 performs the function for its particular stream).

I will shortly release more info about the protocol, and plan to give a talk on it (hopefully with a fully fuctional MacPacket demo) at the Southnet Packetfest in Atlanta next month (Y'all come now, hear?).

This is enough for tonight. More later. 73, Jack B.

C2987 CC224 Lyle Johnson (WA7GXD,2973) 10/24/85 1:54 PM L:34
KEYS:/RS232 FLOW/STA LED CURED/EGGS DELIVERED HERE(FPOR MY FACE)/

Howie,

Oops! The STA LED problem was a release 1.1.0 problem. Eric verifies that it isn't a problem with release 1.1.1. This was my misunderstanding, my apologies.

Here is the straight dope on the flow control problem (Eric is sitting at my right elbow as I key this in).

During power up, while the CON led is illuminated, the TNC 2 sign-on message comes out normally (baud rate to computer is 1200 baud). When the CON led extinguishes, output originated by the TNC ceases. Characters typed in echo normally. Since he is in cmd: mode at power up, any command issued is executed normally, but the response indicating the command was acted upon (eg, value was xxxx) does not appear, nor does the command prompt (cmd:).

If he then connects to a packet sataion (including a self-connect), and the CON led comes on, then all the command mode information that didn't appear as per above, suddenly dumps to the screen and the unit oiperates normally.

Upon disconnect, two asterisks appear, then nothing (except echoed characters) appears until the next CON led illumination.

Eric is using a Radio Shack Color Computer (says color but the case is mostly black and white) with a 3-wire interface -- txd, rxd and ground. There is no othe rpin connected at either end of the rs232 cable. This is the same cable connection that he has used sith TNC 2 during beta test, with other TNC 2 software, with TNC 1 and with the Beta board. All of which work without displaying this characteristic.

End of report. Keep the caffiene handy. Your faityhful, though sometimes not-totally-accurate scribe,
LJ

C2987 CC225 Howard Goldstein (N2WX,2987) 10/24/85 3:24 PM L:12
KEYS:/DATA LOSS EXISTS/FIX COMING/

to: beta
fm: howie
re: data loss problem found
dist: closed

First Skip thanks for sending the DISP dump along. I have indeed been able to duplicate the CONV mode data loss when ECHO is OFF.

This is a serious bug! I think I have a fix that seems to work and will upload it later today or tonight after class. 73 Howie

C2987 CC226 Howard Goldstein (N2WX,2987) 10/24/85 3:40 PM L:8
KEYS:/SEND EGGS HERE LYLE/THEY BELONG ON MY FACE NOT YOURS/FANTOM FLOW?/
A: 224

Lyle - If Eric can, ask him to check the voltages at U9b pin 3 and U21 pin 23 when the board "isn't on speaking terms". I suspect there's a short that's keeping the U21-p23 pin high when the CON LED is off, and it's flowing off the xmission from the TNC to the CoCo

PS this problem would not have been noticable on earlier versions that didn't recognize hardware flow on default power up

73 Howie

C2987 CC227 Tom Clark (W3IWI,2976) 10/25/85 12:51 AM L:7
KEYS:/OMISSION FROM MANUAL/AWLEN & PARITY SET ONLY ON POWER-UP/

Had a user (our friend WB6RQN) report he was unable to change PARITY and AWLEN. I recalled that these parameters were set only on power-up and was going to reference the manual to him. Scoured the manual high and low and couldn't find ANY reference to it! Looks like this should be flagged in subsequent errata! Howie -- is there any real reason for not allowing AWLEN and PARITY as immediate commands? Might be easier to fix the code than to fix the documentation!

C2987 CC228 Lyle Johnson (WA7GXD,2973) 10/25/85 1:06 AM L:5
KEYS:/MISCELLANEOUS INFO/SIGH/

unfortunately, the eproms are burned, kits are packaged and errata is printed. I can do another errata, but I think the 1.1.1 code we have now is the release stuff we will have to use. kits start shipping again tomorrow or monday at the latest 9sigh). I will have eric check voltages. (errata runs 13 pages this time!). Lyle

M 7139 Harold Price (NK6K,2972) 10/25/85 5:18 AM L:151
KEYS:/HOW MANY TNCS ARE THERE?/
TO: (Group 95)

From: Harold Price, NK6K
Dept. of Prognostication
To: ALL
Re: Fun with numbers
Dist: Open

How many TNCs are there? (And how many will there be?)

Aside from being an answer to a trivia question, it is one of many items that must be considered by network planners.

The installed base, as of the end of October, 1985, by my estimate is: 8750.

This number is based on actual TAPR numbers (3700), and information from other vendors. The other vendor numbers are a blend of (1) their numbers (high), (2) their competitor's estimate (low), and (3) various other information sources. Results are checked for credibility against published packet census data from various locations. The number includes TNCs shipped overseas, but does not include TNCs manufactured offshore, nor does it include bare board TNC-1s manufactured and sold worldwide under TAPR's OEM agreements.

These numbers are within 15 percent, and are conservative. It isn't sporting to break out the list by vendor.

Other interesting facts about these numbers:

An estimate of the money spent on TNCs so far is $1,892,000.00 (One million, eight hundred ninety-two thousand dollars).

The Heath and Kantronics units have only been on available since April 1985, the TAPR TNC-2 since August 1985. A little more than half of all TNCs in existence have been sold in just the last 6 months.

How long have TNCs been around?

Some major amateur digital milestones.

Sept 1978 non-baudot digital transmissions made legal in Canada. Digital experimentation begins.

Jan 1979 VADCG group formed. This group produced the VADCG TNC, some are still in use today.

Summer 1979 Work beings in Ottawa and Montreal. Total North American digital usersj Ness than 30.

March 1980 Ascii data legalized in U.S. Canadian missionaries armed with VADCG TNCs and software cross the boarder.

Dec 1980 First U.S. digipeater goes on the air in San Francisco, it uses homebrew hardware and software based on the VADCG protocol (now called V1).

1981 First great packet diaspora begins. VADCG distributes PC boards. Homebrew systems are developed. Most areas standardize on 1200 baud bell 202 modems and VADCG compatible hardware. Locally maintained software versions in San Francisco, Washington DC, Vancouver, and elsewhere begin to diverge.

Oct 1982. AMSAT and AMRAD host another in a series of meetings to solve the divergence problem by developing a protocol standard. Other major goals include the desire to support more than the 32, 64, or 128 users allowed by then current V1 implementations. The AX.25 standard is born. Total North American digital users: no more than 200.

Jan 1983. After several months of design and testing, TAPR produces 170 assembled and tested TNCs.

Oct 1983. TAPR TNC kit (now called TNC-1) is beta tested by 19 users.

Dec 1983. 200 TNC-1 kits are shipped. In the mean time, more VADCG boards were assembled. GLB takes out first ad in QST for an assembled and tested unit. Total TNCs: about 650.

1984. TAPR begins to ship TNC-1 in bulk. They ship an average of 120 TNCs/month for the next 15 months. AEA announces an assembled TAPR TNC-1 clone at the Dayton Hamvention. Packet hits the big time when

Lyle, WA7GXD wins the Dayton Technical Excellence Award, he accepts on behalf of packet radio and TAPR. AEA legitimizes packet by placing the first full page ads in the big ham magazines. At the end of 1984 there are more than 2500 TNCs.

1985. Heath announces HD4040 TNC-1 clone kit. Begins shipping in April, sells out first 500 in three weeks. Kantronics announces "Packet Communicator". TAPR announces TNC-2. GLB announces PK1L. AEA announces PK-64 and PK-80. For a time in August, most of the packet industry is "sold out", with demand far exceeding production.

Summary:

1982	200 tncs.	
1983	650 tncs.	
1984	2500 tncs.	
1985	10000 tncs.	(projected, 8750 tncs through Oct 1985.)
1986	????	

Where will we be in 1986?

AEA has announced the PK-64. Targeted are all those C-64s setting in closets out in Ham-Land, it is the first low priced unit to include Packet, RTTY, and AMTOR in the same box. Industry-wide, expect at least another 5000 TNCs by the next ARRL networking conference in March, 1986. That's about 14,000 TNCs.

How far can we go?

The 1985 Callbook lists about 460,000 amateurs in North America. A study commissioned by the ARRL a few years ago found that half of the amateurs polled considered themselves "active". Perhaps a better indication for our purposes is the number of RTTY units sold during the big computer-RTTY boom of a few years ago. A discussion with two vendors at the Dallas hamvention early this year and with another vendor this week resulted in a guess of 50,000 units sold to 30,000 individuals.

I believe our growth will peak somewhere between this figure and the total number of 2 meter HTs. I can't get a good guess for that last figure, anyone want to take a stab?

I also believe that for Amateur Radio to survive in the long run, we'll need new blood. A large percentage of new hams will be interested in digital radio, many will be drawn to our hobby solely for its digital aspects. So in the end, who can say how far we'll go?

Next issue: What has this got to do with network planning? or
Network gateways: love 'em or hate 'em, we'll have 'em.

M 7422 JEFF WARD (ARRL,2977) 10/25/85 10:21 PM L:11
KEYS:/SOFTNET/
TO: (Group 95)

RE: Softnet and their fast modem

The only sources of information that I have ever had on the Swedish SOFTNET group were therir newsletters. Their address was in a couple of issues of Gateway, and I am sure that if you send them a letter they will repl.y. They have kits available for complete SOFTNET nodes, and I think that they are also offering the nodes (which include radios) assembled and tested for a reasonable price. I am surprised that there has been no experimentation with their stuff in the U.S. Any FORTH maniacs out there?

C2973 CC31 Harold Price (NK6K,2972) 10/25/85 8:19 PM L:170
KEYS:/MAGIC BOXES/

From: Harold, NK6K
To: layer three group.
Re: network access via magic box
Dist: open

Its time for my two cents worth, even at risk of being called the captain of the don't care crowd by Phil.

Unfortunately for the theoretical bunch, amateur radio is the real world. It is in some ways a bit more real than the bastions of TCP/IP or X.25, since we don't get large infusions of public money or that dirty "profit" money. We have to pay for our resources dearly, and employ large amounts of slave labor.

Packet Radio in interesting in that it has two groups of participants, those who are interested in the data that is sent, and those who are interested in the way it is sent. I've discussed this before. Each group supports the other. This gives those of us in the network planning area a problem, one of supporting "users" while playing with the network.

While some amount of arguing and theoretical gnashing is good, there are some realities being ignored here.

Here is one such reality:

```
Fig 1.
Unmodified -(AX.25)--Magic Link---(backbone net----Magic Link -(AX.25)--Unmod
AX.25 TNC            Box          protocol)        Box                    TNC
```

which some of you will recognize from one of Phil's messages.

There are now, and will always be, more "unmodified AX.25 TNCs" than anything else in the network. The label should actually be "unmodifiable AX.25 TNC", because there are too many of them to update easily. In another message I pointed out that there are 8750 TNCs today, with at least 14,000 due by the 5th ARRL networking conference in March, 1986. 14,000 TNCs aren't going to have a network layer and a transport layer added to them any time soon. If we want the majority of users to be able to connect to the network, they're going to have to access it through a network entry node, or "magic link box".

Some people have said that figure 1 is undesirable, and to some extent I agree. In a perfect world it would be nice to have each individual directly connected to the network. Some people have gone further and said that figure 1 is unrealistic, unreliable, unimplementable, or outright criminal. I disagree. If we change some of the labels on Fig 1, we get this:

```
Fig 2.
dumb terminal--------Magic Link---(backbone net----Magic Link-----------HOST
                     Box          protocol)        Box
```

a further mapping results in:

```
Fig 3.
your terminal------telenet node---(telenet)-------telenet node----------EIES
```

Not only is Fig 3 (and therefore fig 1) implementable, it got you this message. Yes, this is an old way to do things, yes, there is no end to end guarantee that data sent from EIES will appear on your terminal. But it does the job. And given the realities of packet today (8750 unmodified TNCs), we'll end up implementing Fig 1 anyway. Is that bad? Read on.

The biggest problem I see with the theoretical blasting going on now is a lack of a defined goal. What kind of a network is envisioned? It appears that

```
Fig 4.
end user tnc ------------(network)--------- end user tnc
                                                or
                                                host
```

is the implied goal.

Laudable, but its not within our grasp if we intend to support the user base any time soon.

Fig 5.

```
TNC1                                              TNC7001
.                                                 .
.----------------------(NETWORK)-------------------
.                                                 .
TNC7000                                           TNC14000
              |                        |
       interface A                interface A
```

In fig 5, 14000 TNCs (march, 1986) must be updated, and kept up to date, with protocol and interface a. This is unlikely in the extreme. To be fair, it is also taking the fig 4 model to the extreme, but it seems that this position has been expressed lately.

Fig 6.

```
TNC1                                                             TNC7001
.                                                                .
.----------(NETWORK node)-----(network)-----(network node)--------
.                                                                .
TNC7000                                                          TNC14000
         |                 |             |                   |
    interface B      interface A     interface A        interface B
```

At last year's meeting of the digital committee, the group decided that fig 6 was more likely because interface B could be fixed at what was available (AX.25 on 14,000 units), and interface A could be diddled around at will because it exists on far fewer units.

Someone recently lamented "Woe is us, how will we tell the magic link box who we want to talk to?". On the network you used to read this message, interface b is an ascii serial stream, and you used x.3, x.28, and x.29 to get the network to do your bidding. On our network, interface B is an AX.25 connection, X.3, X.28, and X.29 can be used to handle PAD control.

To make the point again, it has been argued that fig 6 (a mapped fig 1) is unworkable. It got you this message, and it is far better than a chain of digipeaters. And it's a good thing that it is implementable, because with an installed base of 14,000 by March we haven't got many other choices than to offer network service via a gateway.

I'm not advocating a freeze on network access development to a level two interface and a network diagram as in figure 1. But for several reasons it is the easier path to take first:

o User pressure. Users, including us, need a more reliable bulk file transfer system than we have now. The simulated earthquake and the real fires in southern California have shown the limitations of our current system. There is real need for something now.

o User inertia. There are 8750 TNCs now, 14,000 by March. The majority won't be upgraded to layer3/4 boxes soon, maybe never.

o Staged growth. Fig 5 is an all or nothing proposition. Fig 6 allows for distributed development and partitioned growth. The network will grow bit by bit, with need forcing innovation. Sure, more energy is expended in the long run, but the increments are manageable. Amateur Radio can't afford to "buy" our network outright, we can only afford the installment plan. It therefore makes sense that we design to a model that will work by installments.

Note that this discussion has not touched on VC vs datagrams. The final choice will be invisible to most users anyway. As time goes by, hosts will connect directly to the network rather than through a gateway node. Users will too, but not soon, and not most of them.

Summary.

The purpose of this document has been to counter some of the comments that figure 1 is undesirable or unimplementable. It discusses figure 1 as inevitable based on current user numbers and devices, and desirable because of the possibility of incremental implementation. It uses TELENET and EIES as an example of a figure 1 implementation that provides a data transfer service far better than what we can now offer with a string of digipeaters, even though it does not maximize "elegance".

C2973 CC32 Lyle Johnson (WA7GXD,2973) 10/25/85 11:53 PM L:38
KEYS:/NETWORKS/USERS/NUMBERS/

Of user interfaces and the "majority" of users.

It is true that most TNCs are connected to computers. This is NOT because most people now coming into packet radio are dedicated computer types. It is because a C64 is cheaper than a VT100.

A LARGE portion of TAPR's mail consistsa of: What is RS232? How can I connmect my VIC 20 to a TNC? What program do I use to talk to the TNC? What is ASCII? What is BASIC? Do I need to know how to program to use packet radio? etc.

Go to any ham radio conevention where a couple thousand people attend. Hang around the packet booth. Ignore the 8 people that spend the whole convention at the booth (no, not ignore them, but ignore them for now...) and you will see that the vast majority of the gawkers and enquirers are beign handled by the other 3 people at the booth who are not part of the 8 who engage in hard-core technical discussions. And the 3 people are answering the aboce questions for joe and mary HT.

And Joe and Mary, the ones that hang out on the locasl 2-meter machjine and talk about being "nearly destinated" are the same Joe and Mary that are going to be the largest percentage of user types in any packet network in the next year.

The hi-tech typess are already saturated in packet. Most of you have more than one TNC and many of you have been involved in packet since the early days of 1982 and before. You understand the issues, or probably do.

Joe and Mary want to communicate. "I am nealry destianted, 10-4." will be the majority of the error-free traffic ahndled.

Think about it. 10,000 today (well, 10,000 by the middle of Nocember), 14,000 by ARRL number 5. 25,000 by Christmas 1986. maybe more. probably not less. Most with C64s.

Fig 6, here we come!

Lyle

C2973 CC33 Harold Price (NK6K,2972) 10/26/85 1:10 AM L:127
KEYS:/NETWORK PLANNING/

To: Level 3 group
Fm: Harold, NK6K
Re: Just what are we doing?
Dist: open

It strikes me, whilst perusing the recent goings on, that I don't have a clear idea of what sort of problem is being solved here. When most organizations sit down to begin network planning, they have a goal in mind. "We're going to connect the 4 PCs down the hall with the printer here and the bar code reader over there", or "we're going to connect the central office with the 14 district sales offices", or "We're going to wire the new building so that each of the 100 currently planned user stations can transfer files and mail between each other".

The amateur thinking in some parts seems to be

> We need to connect computers
> Therefore, we need a network
> (blank) is a pretty good network so
> let's implement blank.

Thus, we get offers to get sent copies of the Complete ARPANET Standards, or the X.? fascicles, or whatever. We get comments like "We're doing ethernet without the cable". I'm guilty too.

That leaves us where we've been for 3 years. Trying to get an entire network standard to be accepted and implemented in one fell swoop.

It's hard to not want it all. Its hard to talk about the amateur network to your non-ham professional peers. They say

> "(sniff), protocol (blank) has solved all of the problems you are ever likely to encounter. We here at (blank) believe that the (blank) protocol is best for all needs, and certainly best for your piddling network. If you aren't pushing for (blank) to be implemented you are uninformed, misinformed, microcephalic, or worse."

The protocols we're talking about now, if fully implemented, will allows all manner of wonderful things. Process to process communication between two hosts. Digital Voice. Dynamic routing. But do we need it now? Will we have any process to process applications by 1987? Digital voice? More than one path between LA and Colorado? Maybe by 1990, but not by 1987.

Can we afford to "pay" in 1985 for things we won't need until 1990? In an earlier paper, I postulate no. All we must do is make sure we don't preclude addition of these elements at a later time.

What do we need to do between now and 1987? I submit that we need one thing and one thing only. File transfer.

Direct DX TNC to TNC connections are too expensive for our fledgling network. At best, there are 12 2 meter hops between San Diego and Vancouver. Some of those paths will be marginal at 9600 baud, so let's say 12 hops at 1200, and 15 hops at 9600. Assuming no overhead, that's a rate of 100 baud or 640 baud, depending on the technology used. All the way to Vancouver at 9600 by 1987 is a little iffy. Add in keyup delays, aloha effect, and acks (of whatever type), and we're talking a realtime date rate of less than 20 baud for 1200, and 100 baud for 9600. I think it is safe to say that we won't have a lot of person to person real time chatting going on between San Diego and Vancouver any time soon. Don't disallow it, just don't optimize the network for it over store and forward files.

Are we blessed with several paths to take on our winding way up the coast? Nope. Not by 1987.

Eventually we'll have several choices, 56kb backbone San Diego to San Francisco, 9600 through far northern California and Oregon, maybe 56kb between Seattle/Tacoma and Vancouver. 400 baud through Phase IIIc. 19.2kb through Phase 4A2. But not by 1987.

What will we have? Several centers of activity that want to exchange information in the form of files (much like this one). A fire camp in a canyon that wants to get health and welfare traffic out into the NTS. People using spare capacity to talk real time. Not much digital voice.

And not much bypassing of down nodes on a realtime basis either. We need some reports on downtime figures for digipeaters. MTTR, MTBF, what failed (radio or TNC), etc. It has been our experience here on the lower west coast that when they're up they're up, and when they're down they're down for a long time.

Digipeaters do go down, but the number of state changes is small. Long paths that fade in and out won't be part of a 9600 baud backbone so we need not worry about that by 1987 either.

As I've said before, now and in 1987 there aren't many choices in any case. So discussion of the best failure detection schemes and multiple path routing schemes are premature. As long as we pick a network implementation method that doesn't lock us in.

That's why I like this picture:

Fig 1.

```
Unmodified -(AX.25)--Magic Link---(backbone net----Magic Link -(AX.25)--Unmod
AX.25 TNC            Box          protocol)        Box                  TNC
```

Today the magic link box supports fixed routing, because there is only on path anyway. Tomorrow it has externally specifiable routing decision tables. Next make it autonomous. Today it has no provisions for fast forwarding of non-error detected packets. Tomorrow it allows digital voice.

Lyle's NNC design allows us to take the first step, and fill the Magic Link Box slot. It has enough capacity for what we need now, and what we need in 1987. It is expandable. It also has the advantage of several people who have lined up to write code for it.

So that's the kind of network I think we need (in 1987), and the kind I'm working on. What sort of network are the rest of you working on? What attributes does it have, what problems does it solve?

NK6K

C2987 CC229 Howard Goldstein (N2WX,2987) 10/25/85 6:41 PM L:42
KEYS:/RECONNECT PATH BUG/W2VY/

M 7271 Thomas A. Moulton (W2VY,995) 10/25/85 2:14 PM L:37 KEYS:/TNC
2/FADPAD BUG/ TO: N2WX, N2DSY, John

Howie,

I think I found a Bug in the TNC 2 code! (yea yea sure you did, when are you gonna learn how to use it VY???)

I tried to connect to the bbs I use via a poor path 'cause it sounded quiet I then did a double DISCONNECT and re-connected via a better path, I finally got connected, but the path was strange, watch this...

cmd:C WA2SNA-1 V WA2SNA-2 [a couple of tries later] cmd:D cmd:D Retry Count exceeded ***DISCONNECTED*** C WA2SNA-1 V WB2VTN-1 [a few tries] ***CONNECTED TO WA2SNA-1 V WB2VTN-1 ***

Looks good HUH? well just listen to the audio and you will hear that when I SEND it goes Via VTN and when I RECEIVE it goes Via SNA-2 AIN'T THAT WIERD???

The problem is that the paths are not checked to be correct there are 2 options, either verify the entire path to be the same or use the path in the UA connected frame

If the verify of the entire frame fails then send a DISC to via the old path and then when you retry it will make it...

it really sounded funny!

Howie, this ain't too clear, data went from VY to VTN-1 to SNA-1 the ACK went SNA-1 to SNA-2 to VY

lemme know what is still muddy

M 7516 Gwyn Reedy (W1BEL,2975) 10/26/85 7:39 AM L:4
KEYS:/SWEDEN/
TO: (Group 95)

Den Connors, KD2S, visited the Swedish group and has been in contact with them subsequently. Den might be able to provide more information, if you can get hold of him. Gwyn.

C2973 CC34 Jack Brindle (WA4FIB,2963) 10/26/85 2:46 AM L:85

This is an interesting view being re-presented now. It basically says, lets concentrate on the basics and figure out ways to make file transfers more error free. But, let's leave the TNCs out of the networking system. It sounds very interesting, and is close to what many packeteers are calling for. I would ask just one modification. If I happen to have a TNC/Computing system that speaks the same protocol as the "black box" network, then by all means let me tie it in. Maybe not at 56KB (or even T1 rates), but even at 1200 baud it gives me advantages in transferring whatever data we need. The big question we must ask is what will be using the system for, and when. Well, digitized audio is here. It is almost trivial to digitize voice and send it across the wires. Ah, but we have voice radio for that. Packet does provide the unique ability to send audio in a store and forward mode, to be played back whenever the receiving operator desires, however many times he wishes. So what else? How about pictures? MacPacket will be able to send digitized pictures very soon. Sure, initially they must first be saved in a file, then sent as a file transfer. But it won't be long before MacPacket will be able to send the digitized pictures straight from the camera & digitizer. This is happening NOW! Still not convinced? Imagine an interactive game on packet involving many users being quickly updated from a central station. This type of gaming occurs every day on computers & networks all over the world. An example is the MultiTrek game available from DECUS to run on the PDP-11 and VAX. Why not bring this to packet radio? These things require high speed communications between many users, something that a real network is uniquely positioned to provide. The desire of traffic handling through packet across the nation (one of our major justifications) will not happen with the present system without additions to extend our coverage range. The WORLI BBS is one of these extensions, providing non-real time network messaging services.

Essentially, what I am advocating is a system that allows present TNC users to work side by side with those running more advanced systems. It should handle the present users well, but offer additional services to those of us with advanced hardware and software.

There is an additional consideration in our efforts to build a system. The TNCs and controllers we design are finding their way into the commercial market. Ask AEA where their PKT-1 sales efforts are directed. The commercial market is ripe to literally explode into packeting. What is stopping them? Two things, a viable packet communications system, and government regulations. The second is far easier to handle (yes I know about red tape) than the first. This is why we must carefully design our networks.

Another question worth asking is whether packet is simply a fad. CB, video games and home computers have so far fallen into that category. Could packet be a fad for ham radio? The availability of cheap TNCs parallels similar events in the three "fad" markets previously mentioned. We will undoubtably see a major price war as the vendors try to capture large market shares to make their money quickly before the market goes away. Let us hope not. The good effect on this is while it is happening it gives the system builders extra impetus (and funds) to build the systems. Let us use it well.

Now, for the file transfer functions. Those following my long monologs have noted a new session protocol. It is designed to enhance file transfers and QSOs alike. Combined with a machine-dependent file protocol (in the case of MacPacket, that is MacBinary) it will allow any kind of file to be transferred from one computer to another, and be duplicated on the receiving system in a state identical to the way it existed on the sending system. This means, for example, that MacPacket users can send an application from one Mac to another, then immediately execute the program. It is an extremely versatile protocol. The penalty for using it? about 5 bytes per packet. It's advantages far outweigh that. So when will you see it? Be patient, the Southnet Atlanta Packetfest is only four weeks away.

Now for a plea, directed at Harold, Howie, and anyone else designing TNC software. It has been noted that most TNCs are connected to computers. It is also interesting to note that there are many hams running PCs and Macintoshes, which cost more (but do more) that the Vic 20 and C64. To adequately interface the computer to the TNC, we need two things. The first is a decent hardware connection. We have that in the RS232 wiring. The second is good computer compatible TNC firmware. It is unreasonable to require a computer wade through verbose text to collect information that should require one character. WA8DED has made a good effort at designing such an interface. Please consider adding something similar to your code. An example to follow is the method used to interface microprocessor peripherals to a mpu. The devices generally have a command/status port and a data port. The mpu must specifically ask the device for status or data, it does not just volunteer it to the processor. This interface has proven itself time and again in many environments, not just computers. As I would envision it, the TNC would have two modes of user interface. One very terse, for computers. The other verbose, for humans. Let me provide my own style of verbosity, don't force it on me.

Thanks for listening. We must be doing something right, look at all the attention we have gathered. It continues to amaze me that hams still have the lead in packet radio, far and away ahead of the commercial outfits. I wonder, though, how many commercial concerns have approached TAPR for help in establishing their own commercial networks. My own MacPacket has drawn inquiries for many interesting commercial uses. Like I said previously, the commercial packet world is about to explode. It will be interesting to see where the mode we have pioneered will take us!

73, Jack B.

C2973 CC35 Phil R. Karn (ka9q,2979) 10/26/85   5:57 AM L:59
KEYS:/COMPATIBILITY VS PROGRESS/

Harold, thanks for your comments, they are very well taken. I could take a rather cynical view and say that they prove beyond a shadow of a doubt that amateur packet radio has grown far too fast for its own good. However, if I read between the lines I get the sense that we aren't fundamentally at odds, only that we disagree in our short term priorities.

First of all, we seem to agree that our ultimate goal is a "homogenous" network. Second, we agree that ad-hoc kludges (such as our means of accessing EIES) CAN be made to work in a limited fashion (although if you were accustomed to something that works much better, you might also consider it far from satisfactory). Third, we agree that there are a lot of TNCs out there, (and I might add that people will invent ad-hoc kludges no matter what we do.)

However, at this point I must take issue with some of Harold's remarks. He points out that packet radio is growing rapidly, and that this is an argument for supporting "the unmodified TNC". I would respond that if packet radio is growing this rapidly, then we had better fix things ASAP, because it'll only get a lot harder as time goes on. I know that burning 14,000 sets of EPROMs is a pretty horrifying thought, but aren't these TNCs ever going to have software updates? I can't imagine that the thousands of TAPR TNC owners out there are going to run version 3+.whatever from now until doomsday. Nor do I hope that all the bugs present in other manufacturers' TNCs will remain forever unfixed. And it takes the same amount of effort to burn and distribute ROMs containing a set of minor bug fixes as it does to burn and distribute ROMs with major new features (like support for TCP connections as well as vanilla AX.25).

Amateur packet radio currently suffers from a severe lack of long range planning. The sooner we could get our act together, agree on a network and transport protocol and indicate to the manufacturers and to the general amateur community what our plans are, the sooner will come the day that every TNC will implement it. Yes, "magic boxes" are probably inevitable as backward compatibility aids. But we had better keep our ultimate goal in mind and stress to everyone the temporary nature of the intermediate steps. If we don't, we're liable to end up with "magic boxes" (and probably worse) as permanent warts in our network and don't think any of us want that.

Harold says that amateur packet radio is the "real world", at least in the sense that we don't have a lot of money to waste. Absolutely! THAT'S why it's so important that we look a little farther ahead than our next PC board design and stop wasting our time and money on stopgap solutions. There once was a time that the "bearded experimenter" types could get together and discuss fundamental issues like protocols, addressing and routing techniques and really get some good ideas flowing. Unfortunately, things suddenly became "real" before we knew it (and before we had finished the groundwork discussions), and now many of the "bearded experimenter" types find that they need 30 hour days just to talk at hamfests, write introductory articles for the masses, and (at least supervise) the stuffing of parts into hundreds of boxes for UPS shipment. It's certainly understandable that these people have little time or energy left to think about the relative merits of TCP/IP and X.75. As the story goes, when you're up to your ass in alligators, it's very difficult to remember that your original objective was to drain the swamp.

Phil

C2973 CC36 Phil R. Karn (ka9q,2979) 10/26/85 6:01 AM L:136
KEYS:/KA9Q PROJECTS/

I'd like to describe to this group what my current plans and projects are.

1. I'm currently debugging a TCP implementation written in C. Compiled on an 8088 the object code is a little over 5K bytes. I'd like one of the TAPR software wizards to take a look at it and see what it'd take to add it to the TAPR code, along with an option for initiating a "raw" AX.25 connection or a TCP-on-top-of AX.25 connection? If this works, it might take care of at least a few thousand of those 14,000 TNCs.

2. I've had the BRL IP gateway code sitting here for a month or two. (Note: in Internet parlance, an "IP gateway" is synonymous with a packet switch that switches IP datagrams. This is not to be confused with a "protocol conversion gateway".) This code is in C and was originally intended to run on an LSI-11, a machine with a 64K address space. It speaks the EGP (Exterior Gateway Protocol), the standard routing protocol on the ARPA Internet, and therefore provides automatic routing.

Assuming that the object code size is the same when compiled for a Z-80, I will try to bring this package up on a Xerox 820 (some space can be freed by deleting the drivers for such devices as Ethernets and high speed fiber links). I will attempt to integrate this code with the AX.25 level 2 implementation I've already written, and will add support for the "serial line IP" (SLIP) protocol that allows IP datagrams to be sent over asynch interfaces. With a FADCA card, the result would look like this:

```
-------------------
|   IP gateway    |
|     code        |
|-----------------|
| SLIP  |  AX.25  |
-------------------
   | |     | |
   ^ ^     ^ ^
  asynch   synch
```

The IP level packet switch will therefore see four ports: two happen to be synchronous and are of course connected to radio modems. The asynch ports could be used in the following ways:

1. Connected to a local "client" host speaking TCP and running some sort of application, such as a PBBS. Since my TCP is fairly portable, and SLIP requires only an asynchronous interface, this means that a lot of machines could speak end-to-end TCP over packet radio without requiring either an HDLC interface on the machine itself or TCP to be implemented on an external TNC.

2. Connected to a local "client" host programmed to act as a TCP "PAD":

```
                           asynch to
                           dumb term
                               ^
          ----------           |
          |terminal|           |
          |--------|       ----------
          |Cmd Int |       | Cmd Int|
          |--------|       |--------|
          |  TCP   |   or  |  TCP   |
          |--------|       |--------|
          |  SLIP  |       |  SLIP  |
          ----------       ----------
              |                |
              ^                ^
           asynch           asynch
             to               to
            IP gw            IP gw

             (A)              (B)
```

"Cmd Int" is Command Interpreter, the thing that you find in every PAD which prompts you for a destination address and does other PAD-like things.

Note that version (A) is already available off-the-shelf, FREE, in the form of the MIT PC/IP package. Besides allowing remote login (i.e., PAD services) across the internet, this package supports reliable file transfers (did I hear somebody say "file transfer" earlier?) with the upper-level TFTP protocol.

Since IP is spoken over the serial lines to the IP switch, there is no need for any form of hardware flow control on the serial lines. This is handled end-to-end by the TCP's, and should the buffers in the 820 occasionally overflow the TCPs will take responsibility for retransmission.

3. Connected to another IP gateway, either local or across a nailed-up phone line. The other gateway could be another 820, providing access to two more radio channels, or it could be a UNIX machine with access to the Internet...

Note that if this works on the 820, it should also work on Lyle's NNC.

Now to the subject of backward compatibility. Since I guess I'm resigned to having an AX.25-to-TCP protocol conversion gateway in here somewhere, (at least temporarily) it could be built on another 820 or NNC, programmed like this:

```
-------------------
|     Cmd Int     |
|--------|--------|
|  TCP   |        |
|--------| AX.25  |
|  SLIP  |        |
-------------------
    |        |  |
    ^        ^  ^
 asynch     synch
```

Note that AX.25 here is shown at the same level as TCP, with the "Cmd Int" layer connecting the top of TCP with the top of AX.25. This is the canonical protocol conversion gateway. Logically, it should really be folded outward like this:

```
synch <-------|AX.25|<--|Cmd Int|--->TCP--->SLIP--->asynch
```

This is basically the same thing as my TCP PAD shown earlier, except that the terminal interface on the top is now replaced with an AX.25 module "turned upside down". A user would connect with a dumb AX.25 TNC to the AX.25 side and then converse with the TCP command interpreter. As long as things are kept simple (i.e, no more than one session at a time, and so forth) this will be a doable project -- but it will be limited in functionality, and as I said earlier, meant only as a stopgap.

Another possibility would be to take an ordinary AX.25 TNC and connect it to the serial port of PAD version (B) shown earlier; of course, this could only support one user at a time.

These are just a few examples on how the various software modules I'm writing or obtaining could be fit together. If we had a machine large enough, all these functions (IP packet switch, AX.25 link control, incoming AX.25 "user" ports, TCP "pad" functions, BBS services) could be put in one place. A 68000 can easily do this and in fact 4.2BSD (which runs on 68Ks) already provides everything here except the lower level AX.25 interfaces -- but X.25 support, both for "dumb PAD" incoming calls and for IP relay, is already there.

Phil

C2973 CC37 Thomas A. Moulton (W2VY,995) 10/26/85 10:23 AM L:5

Harold,
Glad to hear your comments.

Phil,
I think we have.

C2987 CC230 Tom Clark (W3IWI,2976) 10/27/85 12:25 AM L:22
KEYS:/BLACK-HOLE MODE/ONE SOURCE DISCOVERED/INVOLVES CANPAC CHARACTER/

Possible "black-hole mode" (BHM)

A corrupt disk file found one way to create a BHM which drove me up the wall for a couple of days. On the 'IWI BBS, I kept finding the TNC2 croaked each morning. After reviewing the logs, I found that it always occurred when the same file was being downloaded. The file did not cause the TNC1 on the other BBS port to croak. I investigated the file and found that in one spot some garbage had gotten embedded. Buried within that garbage was a ctl-c followed about 20 bytes later by a ctl-y. Reading the TNC2 book, I found that the TNC2 had done just what it was advertised as doing. Ctl-C sent the TNC into cmd: mode, where the CANPAC character (default ctl-y) acted as a toggle to freeze all output from the TNC (chapter 6, page 14, last 2 paragraphs). I don't think that this was the BHM I reported previously, but who knows? It is possible that the over-run condition I described earlier could have overrun the UART creating the Ctl-C followed later by Ctl-Y condition.

Howie, is there any character that really disables CANPAC (like zero)? This looks like a vulnerable mode for RLI BBS's which run in CONV vs. TRANS mode.

73, Tom

C2987 CC231 Howard Goldstein (N2WX,2987) 10/27/85 8:58 AM L:36
KEYS:/BLACK HOLE/CANPAC/CTRL Y/CLOCK/DATA LOSS/
A: 230

Tom -

It's probably a pretty good idea to set CANPAC $00 with the MailBox, and it will really disable the action in both CONV and cmd: modes.

I've reproduced another black hole on filling-buffers, but this mode seems okay (i.e. expected). Viz, after eight packets are queued on an individual stream, no more may be queued until something is ACKed.

Internally this is how information-for-packetizing is handled:

```
1.  data pending packetizing?  if so, goto (5)
2.  process any ^r/^x etc
3.  push the character onto the "packet edit" buffer
4.  complete packet built in the buffer?  No, EXIT
5.  insert the packet into this connection's/link's
      list.
6.  did the packet fit in the list from (5)?
      Yes, flag the edit buffer as empty now and EXIT
      No, Set not rdy flag and next time through, go directly from (1) to (5)
```

Basically what this says is that in CONV mode, editing functions (like ^X) will not work IN CONV MODE while a packet waits in the edit buffer for queuing. Note however that echoing, and CMD: line editing are unaffected by this wait.

Other things: CLOCK- I think I found why the clock was performing so poorly. In fact the clock on my 10.0 RLI is now 2 minutes FAST after being set yesterday afternoon. This poor performance was noted quite early on in the multistream software, I think the cause has been corrected.

CTRL-Y performance: Enhanced ^Y handling so it cancels output on the fly. It impacts on the total throughput of the board since its handled in a real time interrupt and it needs to be tested at high speed before its released like this.

DATA LOSS: Skip's data loss problem seems to be related to - hang on now - echoing, and of all things, the clock. More definitive data later, but it appears that when the echo-processing buffer became full things started slowing down. Enough that asserting hardware flow control was delayed past the overflow limit of the input buffer. 73 Howie

M 8623 Pete Eaton (WB9FLW,2970) 10/30/85 12:28 AM L:29
KEYS:/BACK TO NORMAL NEXT WEEK/
TO: (Group 95)

To: All
From: Pete WB9FLW
Subject: TAPR Phone & TNC 2 Kit Orders
Dist: Open, please give widest dissemination

Chris TAPR's Office Manager is trying her best to ship approximately 700 TNC 2 kits. To add to the problem the label printing program is not working so all addressing and UPS manifests are being done by hand.

To add abit of challange to this "chore" the phone rings...and rings... and rings. Chris cannot do these chore at once or attempt to multiplex them together. So it was decided that the shipment of kits had priority (alot of calls were asking where their kits were). With that decision made it was decided to put the famous TAPR answering system on line to inform folks that called in that Chris was busy shipping kits but would bee happy to take calls after 3:30 P.M. (after the U.P.S. pickup was made). Sounds simple enough right?......well the answering machine refuses to work, it either does not answer the phone or answers says nothing and hangs up, or leaves the phone off the hook (and give you a busy signal all day long.

The TAPR phone will not be answered until all kits in stock have been shipped and/or it's after 3:30 P.M. Chris will have some extra help in the office next week to help get things back to normal with my arrival on Saturday.

Please let folks know! hopefully things will be back to "normal" by next week.

C2970 CC161 Paul Newland (ad7i,2978) 10/30/85 9:07 AM L:54
KEYS:/EMERGENCY USE OF PACKET/

FROM: paul newland, ad7i
TO: DRAGNET
SUBJ: product review
DIST: open
DATE: 85.10.29

The following posting was taken from USENET. I forward it as a courtesy to the net. The opions expressed here are not necessarily those of only the author.
======================

Subject: Kantronix in Mexico - Help! + FLAME
Date: 22 Oct 85 02:47:00 GMT
Reply-To: john@anasazi.UUCP (John Moore NJ7E/XE1HDO)

During the Mexico City earthquake disaster, we tried to get a Kantronix packet interface going in Mexico City for health and welfare traffic. Unfortunately, during the short time we had to test it, we couldn't get it to receive anything. The manual was singularly uninformative about troubleshooting. It appears that the FSK work is all done by a magic chip which has no description in the book and a custom part number. We couldn't find any pin that appeared to have the FSK demodulator output.

Does anyone know how to tune this beast up with a scope, rather than just using the frequency counts? Does this thing work? Is it any good? Does anyone at Kantronix know how to write a technical manual?

+++FLAME ON+++
We wasted several hours of valuable disaster net time due to the sorrowful documentation on this product!
+++FLAME OFF+++

What should I tell Carlos (XE1HC)? Should he throw the thing away or is there a rational way to get it to work?
--

John Moore (NJ7E/XE1HDO)
(602) 952-8205 (day or evening)
5302 E. Lafayette Blvd, Phoenix, Az, 85018 (home address)

C2987 CC232 Tom Clark (W3IWI,2976) 10/29/85 11:31 PM L:12
KEYS:/FOR AD7I + WA7GXD/INFO ON TNC2 INNARDS/FOR USE AS DUAL-PORT DIGIPEATER/

Paul, a request, por favor. I am contemplating adapting KE3Z's Dual-port digipeater code for the TNC2 and need some supplementary documentation. Obviously it is ROM low, RAM high, with RAM starting at $8000. Could you provide info on i/o port mapping so I don't have to fish it out of the schematics? Looks like the only change necessary to do the DPD is to swap the SIO/0 for an SIO/2 (to get separate RX/TX clocks on port B), which entails minor surgery on a couple of pins. Comments?

Lyle, I know that the Hitachi CMOS SIO/0 is (sorta) available. Do they also make a SIO/2 in CMOS?

73, Tom
C2987 CC233 Phil R. Karn (ka9q,2979) 10/30/85 2:42 AM L:13
KEYS:/K9NG + TNC-2/

Last week I attempted to connect my TNC-2 to a K9NG modem board. I discovered that the polarity of the DCD line is inverted from what the TNC-2 expects. After fixing that, I discovered that I could generate frames just fine as long as I didn't connect the RXDATA line from the modem back to the TNC-2. If I complete the connection, the TNC will generate one frame and then refuse to send any more.

During transmit the receive data out pin from the modem carries scrambled transmit data. Ordinarily the worst that should happen is that the HDLC receiver sees a few garbage frames. Why should the TNC behave in this way?

Phil

M 9130 Thomas A. Moulton (W2VY,995) 10/31/85 1:18 PM L:12
KEYS:/TNC 2 UPDATE/
TO: (Group 95)

Here is your TNC 2 Update for Today!

Orders up to 1012 have been SHIPPED! (With a couple of exceptions (say 6))

So we should be seeing them show up starting next Wednesday!

They started going out in the mail Last Friday and Chris will be answering the phone today and tomorrow.

If there was a problem with your order she has called you and left word, you can call to check to make sure that yours went out, but she IS calling so don't need to all (500 of us) call her!

M 11126 Harold Price (NK6K,2972) 11/ 5/85 10:02 PM L:65
KEYS:/TNC 2 REV 2/
TO: (Group 95), KV7B

11/05/85
To: ALL, please give widest coverage
Fm: TAPR
Re: Problems with TNC 2 rev 2.

The first of the second shipment of TNC 2s has hit the field, and some problems have been identified.

The following information applies only to TNCs with an ORDER NUMBER greater than 511, those that say Rev 2 below U21.

Symptom: During the test on page 44, the STA light fails to go out.

Solution: The manual fails to tell you to install U24, the second 6264-L. Remove power and install this chip.

Symptom: The digital loopback test on pages 47 & 48 fails. No CONNECTED message is displayed.

Solution: SOME, not all, of the state machine proms (U6) were programmed incorrectly. To see if this is your problem, perform the following steps:

1) Remove power
2) Remove U5(74HC374) and U6(2716 marked STATE) and place on the black protective foam.
3) Install a jumper between U21 pins 13 and 14.
4) Install a jumper between U21 pins 12 and 15.
5) Check for shorts.
6) Apply power, try digital loopback test again.

If the test succeeds, you have a bad U6. To double check, swap U6 with any other working state machine, either from a TNC 2 rev 1 board, a working TNC 2 rev 2, or the K9NG 9600 baud modem board.

If your U6 is bad, put it on a small piece of the black foam, put the chip and foam in the small box your sockets came in, and send the box back to TAPR. We will return it asap by first class mail.

Symptom: Your front panel is missing.
Solution: No one got one. A sheet notifying you of this was left out of the box. The panels will be sent to you via first class mail by the end of the week.

Symptom: Your board says Rev 2, but the manual says Rev 3.
Solution: Replace Rev 3 with Rev 2 in the manual. There is no Rev 3.

We apologize for these problems. Remember, the preceding applies only to TNC 2 order numbers greater than 511 (rev 2). Owners of TNC 2 rev 1 (order numbers less than 512) will want to be sure to see the next PSR, it contains information on how to upgrade your TNC to rev 2. PSR should be mailed in the next 7 days.

TAPR

M 11141 Tom Clark (W3IWI,2976) 11/ 5/85 10:56 PM L:21
KEYS:/TNC2 REV.2/ERRATA/TESTIMONIAL/
A: 11126   TO: (Group 95)

In Harold's note on TNC2 Rev.2 he referred to U6 as a 2716 EPROM. I believe that U6, the state EPROM, is being shipped as a 27C64 in most (if not all) of the Rev.2 TNC2 kits.

As a small testimonial, I played "beta" site for both the Rev.1 and Rev.2 boards. The subtle changes in Rev.2 really do improve the operation. The all-CMOS design plus the mounting of the 7805 eliminate a heat problem experienced with the Rev.1 board. Grounds and power supply traces and bypasses have been improved to improve noise immunity and cut down on RFI. A fifth LED for PTT has been added. And both the ROM and RAM have been doubled; this in turn let our software guru, N2WX exercise several improvements over the earlier software -- things like multiple connections, digipeater call aliases, the ability to re-establish a path thru different routes without breaking the link, etc. are all super!

Those of you with Rev.1 boards will definitely want to make the mods that Harold referred to -- they involve primarily changing the 2 2764 EPROMS for one 27C256, and then changing the (now unused) 2nd EPROM socket to take a second 6264 8k RAM chip -- resulting in 32k ROM + 16k RAM in 3 chips.

73, Tom

C2973 CC38 Phil R. Karn (ka9q,2979) 11/ 7/85 2:28 AM L:43
KEYS:/TCP WORKING!/

I am happy to announce that my TCP code is functional. I successfully carried out its first real workout tonight in a "QSO" with my vax at work. The path looked like this:

IBM PC/XT -- my code, originating end of TCP connection
1200 baud asynch port (using SLIP for datagram encapsulation)
Ven-tel 212 modem
dialup phone line
Penril modem (company modem pool)
Micom data switch
T-1 interlocation link
Micom data switch
1200 baud asynch hardwired line
asynch interface on my office SUN with SLIP driver
SUN Workstation running 4.2BSD; acts as IP gateway
3-Com Ethernet interface
Local area network
3-Com Ethernet interface
Inter-Ethernet IP gateway (another SUN)
3-Com Ethernet interface
Another local area network
3-Com Ethernet interface
VAX-11/780 running 4.2BSD --- other end of TCP connection

The whole setup worked remarkably well. Of course, the slowest part of the path was the dialup phone line. I noticed that the line was fairly noisy. Whenever a SLIP-encapsulated IP datagram was sent over it after the line had been idle for a while, I noticed that it would often not make it through. This is because SLIP does not put any framing characters on the beginning of a datagram, only at the end, and should a noise character occur between datagrams, the datagram that follows will have it tacked on the beginning. Of course, the checksums at the IP level detect this and discard the packet, which TCP later retransmits.

For demonstration purposes, it would be trivial to replace the modem link with a pair of TNCs so that SLIP-encapsulated IP datagrams could be sent over the air without worrying about the AX.25 I-frame boundaries. Of course, since Dave Mills has already beat me to the first TCP/IP contact over AX.25, I don't see much point in duplicating that demo...

Phil

M 11822 Jack Brindle (WA4FIB,2963) 11/ 7/85 9:44 PM L:30
KEYS:/IT'S HERE AND IT WORKS!!!/
TO: (Group 95)

For those of you trying to interface your TNC2 to a Macintosh, be sure to ground pin 1 on the TNC2 DB25 connector. Pin 1 is supposed to be "chassis ground". Unfortunately it is not connected to anything! A lot of cables use this pin for ground instead of pin 7, so it must be tied to ground! The reason for the problems in using the TNC2 with the Mac is that many Mac - RS232 cables use pin 7 of the DB25 for RX data ground, and pin 1 for signal ground. Most, if not all, 103/212 modems have pin 1 grounded to allow this configuration. Also note that the chassis is grounded only through the ground tab on the +5 volt regulator. Anyway, after making this ground connection, everything runs fine! Now I just need to learn all those new commands and parameters. Nice job, Howie!

On another note, I have now moved from Florida to the Atlanta, GA area. my address here is: Jack Brindle 2980 Wayward Drive Marietta, GA 30066. Phone: (404) 565-6002.

More later. I want to "play" with my new TNC2!
- Jack B.

M 11825 Jack Brindle (WA4FIB,2963) 11/ 7/85 9:51 PM L:9
TO: (Group 95)

One more thing - Thanks a bunch to Pete for his help in "finding" my lost TNC2 and explaining the state machine difficulty. Perhaps we should double his salary instead of cutting it...
(Smile Pete. Remember, "Wait'll next year...").

A word of caution, though. One can get addicted to watching the STA LED after you send a packet to see when the ack is received. Also, notice how quiet the TNC2 is - I had to place the antenna of my IC2AT right next to the PC board to get any noise! Nice job, guys!!!
- Jack B.

M 12380 Pete Eaton (WB9FLW,2970) 11/ 9/85 9:44 PM L:21
KEYS:/IMPORTANT INFORMATION FOR ALL TNC 2 OWNERS!/
TO: (Group 95), WB9FLW

To: All
From: TAPR
Subject: Important TNC 2 Information!!
Dist: Open

Feedback from the field has shown two serious problems that ALL TNC 2 owners should be aware of!

If the Lithium battery is not mounted flush against the board and/or the PC board is warped it is possible that the case can be shorted to ground. All owners should insulate the top of the battery with a piece of electrical tape.

Notice to owners of Radio Shack Model 3 and 4 remove R 14, 15, & 16 these resistors have been shown to draw excessive current to the point of damaging the board! Please open your unit and make this modification immediately.

M 12482 Gwyn Reedy (W1BEL,2975) 11/10/85 7:58 AM L:21
(ORIG.) 11/10/85 7:35 AM L:15
KEYS:/PACKET EQUIPMENT ANNOUNCEMENT/
TO: (Group 95)

TO: The Packet Radio Community (Wide Distribution Encouraged)
FROM: Gwyn Reedy, W1BEL
SUBJECT: New Packet Radio Enterprise

.

Andy DeMartini (KC2FF) and I have started an enterprise to support the packet radio community. We will be doing business as Packet Radio Systems in Tampa, Florida. This is a private venture, fully separate from our volunteer work with TAPR and FADCA. Packet Radio Systems is purchasing the OEM kit for the TAPR TNC2 and will very shortly have available TNC2 bare boards, hard-to-find parts kits, full kits, cabinets, etc. Prices will be set with the builder and experimenter in mind. Other products are planned for the immediate future. Watch for our advertisements for details on these items to support the innovators in Amateur Radio.

.

(Your comments are welcome. I intend to continue working for FADCA and TAPR in a volunteer capacity. This new venture won't have any effect on my editorial policy with the publications I edit! Gwyn)

M 12720 Gwyn Reedy (W1BEL,2975) 11/11/85 6:16 AM L:22
KEYS:/PACKET RADIO MAGAZINE/
TO: (Group 95)

ANNOUNCING PACKET RADIO MAGAZINE

.

Beginning in January, the FADCA > BEACON will become Packet Radio Magazine. The publication will feature: o Larger, Clearer type o All pages stapled, booklet style o A two color cover on heavy stock o More pages, more news, more ads o Sections devoted to news from other areas of SOUTHNET and other parts of the country.

.
The purpose of the change is to allow this publication to better serve those readers and potential readers outside of Florida. Organizations smaller than FADCA will be able to have a large monthly publication for their members that includes a locally produced section. Good technical and operational news will be uniformly available to a broader group of packeteers. In effect FADCA will publish a 'group newsletter' for all those organizations that desire to participate, and all concerned will be able to share in the reduced costs of quantity printing and a broader selection of authors. If you think your organization could benefit from participating in this venture, contact FADCA at 812 Childers Loop, Brandon, FL 33511 for further details.

M 13042 Lyle Johnson (WA7GXD,2973) 11/12/85 12:07 AM L:17
KEYS:/OFFICE HELP REVISITED/
A: 11506   TO: (Group 95)

Rich,

.
Regarding the problems you mentioned with the office staff in M 11506.

.
I have given TAPR's EVP a stern lecture on TNC 2 Rev 2's HS-CLIP-1 (an innovation slightly inferior to the ALJ-1000, but still very useful...) and its proper application in Amateur packet radio.

.
I further have docked him two-weeks pay and withdrawn his all-expenses paid trip to the 1st Bermuda Conference on Packet Radio to be held the entire month of January on a lovely island...

.
Thank you for bringing this matter to my attention. It is only through such public 5th column activities that we can truly know the nature of our officials.

.
Lyle

M 13062 Dan Morrison (KV7B,2986) 11/12/85 1:01 AM L:26
KEYS:/CORRECTION TO PSR REV 1 MODIFICATIONS/
TO: (Group 95)

Having found my name cited as one of two authors of modifications to TNC-2 rev 1, I thought, what the heck, I'd try them out. Well, I'm here to report that there's been a slight misprint in the PSR article on upgrading rev 1 to near rev 2 condition. Unfortunately, it's an extremely important error, causing the revised board not to work after revision. Fortunately, it's an easy error to correct if you have been successfully cutting traces already. Here's the correction:

Revision F, Memory modification, leaves out two steps in the conversion of socket U24 to accept a 6264. After steps 6 and 8, add the following two steps:
6a On the top of the board cut the A13 address line to pin 26 of U24. As examination of Fig. 1 of page 10 of the October PSR shows, this trace is easily located as the second trace from the top, within the U24 outline. Be careful not to bugger up any other traces as they are kinda close.
8a In step 8, continue the jumper to pin 28 over to pin 26 as well (i.e., have pins 28 and 26 shorted together by the jumper you install leading back to U25).

These two steps should be performed in the order indicated, as the second step will cause high current drain on the 3 volt battery unless the A13 line has been cut first (ask how I know this...).

Good luck and happy slicing, you rev 1'ers.
73
Dan

M 13100 Gwyn Reedy (W1BEL,2975) 11/12/85 6:35 AM L:11
(ORIG.) 11/12/85 6:27 AM L:10
KEYS:/TNC2 STYLE KITS/
A: 12482   TO: (Group 95), W1BEL

Message 12482, Packet Equipment Announcement, has been modified. The company name is now PacComm Packet Radio Systems, Inc per advice of our lawyer. Also we have extablished some phone numbers. Here they are for your information (and distribution, if you want): Tech info - (813) 689-3523